

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0024
【제출일자】	2000. 12. 29
【국제특허분류】	G06K
【발명의 명칭】	하드 카피 장치용 데이터 압축 및 복원 방법들 및 장치들
【발명의 영문명칭】	Methods and apparatuses for compressing and recovering data for hard copy apparatus
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이영필
【대리인코드】	9-1998-000334-6
【포괄위임등록번호】	1999-009556-9
【대리인】	
【성명】	최흥수
【대리인코드】	9-1998-000657-4
【포괄위임등록번호】	1999-009578-0
【대리인】	
【성명】	이해영
【대리인코드】	9-1999-000227-4
【포괄위임등록번호】	2000-002816-9
【발명자】	
【성명의 국문표기】	보크카레브 세라핌
【성명의 영문표기】	BOTCHKAREV, Serafim
【주소】	경기도 수원시 팔달구 매탄동 성일아파트 205동 1302호
【국적】	RU
【심사청구】	청구
【취지】	특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인 이영필 (인) 대리인 최흥수 (인) 대리인 이해영 (인)

1. 요약서·명세서(도면)_1통

57-2

【요약서】**【요약】**

하드 카피 장치용 데이터 압축 및 복원 방법들 및 장치들이 개시된다. 두 가지의 단계들로 촬영된 영상을 하드 카피하기 위해 사용되며, 메모리에 바이트 단위로 저장되는 소스 영상 데이터를 압축하는 이 압축 방법은, 소스 영상 데이터의 열 바이트들을 행 바이트들로 치환하는 단계 및 현재 압축 처리할 현재 체인의 값과 동일한 값을 갖는 체인이 이전에 압축 처리된 이전 체인들로 이루어진 사전에 존재하는가에 따라 현재 체인과 현재 체인에 후속하는 체인(들)로 이루어진 순차적 체인들 또는 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 소스 영상 데이터를 압축한 결과로서 결정하는 단계를 구비하고, 행 바이트들중 이웃하는 행 바이트들은 이웃하는 메모리 어드레스를 갖고, 열 바이트들중 이웃하는 열 바이트들의 오프셋은 소스 영상 데이터의 행 폭에 해당하고, 압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보 및 헤더에 대한 정보는 (b) 단계가 수행되기 이전에 결정되고 압축한 결과에 포함되며, 체인은 적어도 둘 이상의 연속하는 행 바이트들로 이루어지는 것을 특징으로 한다. 그러므로, 소스 영상 데이터를 엔트로피 디코딩하기 전에 치환하기 때문에 압축율을 개선시킬 수 있고, 템플릿의 크기를 자유롭게 조절하여 압축 속도도 개선시킬 수 있는 효과를 갖는다.

【대표도】

도 1

【명세서】**【발명의 명칭】**

하드 카피 장치용 데이터 압축 및 복원 방법들 및 장치들{Methods and apparatuses for compressing and recovering data for hard copy apparatus}

【도면의 간단한 설명】

도 1은 본 발명에 의한 하드 카피 장치용 데이터 압축 방법을 설명하기 위한 플로우차트이다.

도 2는 도 1에 도시된 데이터 압축 방법을 수행하는 본 발명에 의한 데이터 압축 장치의 블록도이다.

도 3은 치환되지 않은 소스 영상 데이터의 각 바이트가 메모리에 저장된 형태를 나타내는 도면이다.

도 4는 치환된 소스 영상 데이터의 각 바이트가 메모리에 저장된 형태를 나타내는 도면이다.

도 5는 메모리에 각 체인이 저장된 형태를 나타내는 도면이다.

도 6 및 도 7들은 길이가 17인 두 가지의 템플릿들의 예시적인 도면들을 나타낸다.

도 8은 본 발명에 의한 템플릿 결정 과정을 설명하기 위한 플로우차트이다.

도 9는 메모리에 바이트 단위로 저장된 소스 영상 데이터를 체인 형태로 나타내는 도면이다.

도 10은 템플릿 결정 과정에 의해서 결정된 템플릿의 예시적인 도면이다.

도 11은 본 발명에 의한 엔트로피 엔코딩 과정을 설명하기 위한 플로우차트이다.

도 12는 엔트로피 엔코더의 본 발명에 의한 바람직한 일실시예의 블록도이다.

도 13은 압축한 소스 영상 데이터에 대한 본 발명에 의한 바람직한 일실시예의 포맷을 나타내는 도면이다.

도 14는 파일 헤더에 대한 본 발명에 의한 바람직한 일실시예의 데이터 포맷을 나타내는 도면이다.

도 15는 체인 데이터에 대한 본 발명에 의한 바람직한 일실시예의 포맷을 나타내는 도면이다.

도 16a 및 도 16b들은 청크 데이터에 대한 본 발명에 의한 포맷을 나타내는 도면이다.

도 17은 $m=7$ 인 경우 청크 데이터의 포맷을 나타내는 도면이다.

도 18a 및 18b들은 $m=15$ 인 경우 청크 데이터의 포맷을 나타내는 도면들이다.

도 19a 및 19b들은 $m=31$ 인 경우 청크 데이터의 포맷을 나타내는 도면들이다.

도 20a 및 20b들은 $m=63$ 인 경우 청크 데이터의 포맷을 나타내는 도면들이다.

도 21은 제94 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트이다.

도 22는 제96 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트이다.

도 23은 제102 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트이다.

도 24는 메모리에 저장된 데이터의 형태를 도면이다.

도 25a, 25b 및 25c들은 메모리에 저장된 데이터의 형태를 도면들이다.

도 26은 본 발명에 의한 하드 카피 장치용 데이터 복원 방법을 설명하기 위한 플로우차트이다.

도 27은 데이터 복원 방법을 수행하는 본 발명에 의한 데이터 복원 장치의 블록도이다.

도 28은 제302 단계를 수행하는 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트이다.

도 29는 엔트로피 엔코더의 본 발명에 의한 바람직한 일실시예의 블록도이다.

도 30은 전술한 본 발명에 의한 데이터 압축 및 복원 장치들을 포함하는 시스템을 나타내는 도면이다.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<30> 본 발명은 투명 종이(transparent paper)를 프린팅할 수 있는 하드 카피 장치(hard copy filter)에 관한 것으로서, 특히, 하드 카피 장치에서 사용되는 데이터를 압축하고 복원하는 방법들과 그 방법들을 수행하는 장치들에 관한 것이다.

<31> 일반적으로 사용되는 프린트의 공간 해상도(spatial resolution)는 600 또는 1200 디.피.아이.(dpi:dot per inch)로서 그다지 크지 않다. 그러나, 하드 카피 장치의 공간 해상도는 12000 디.피.아이.정도로써 매우 크다.

<32> 이러한 높은 해상도를 갖는 종래의 하드 카피 장치에서 하드 카피를 위해 사용되는

데이터의 압축율은 낮고, 그의 압축 속도 또한 느리기도 하고 가변될 수도 없는 문제점이 있었다.

【발명이 이루고자 하는 기술적 과제】

- <33> 본 발명이 이루고자 하는 제1 기술적 과제는, 두 가지의 단계(bilevel)로 촬영된 영상을 하드 카피하기 위해서 사용되는 데이터를 고속에서 효율적으로 압축할 수 있는 하드 카피 장치용 데이터 압축 방법을 제공하는 데 있다.
- <34> 본 발명이 이루고자 하는 제2 기술적 과제는, 상기 하드 카피 장치용 데이터 압축 방법을 수행하는 데이터 압축 장치를 제공하는 데 있다.
- <35> 본 발명이 이루고자 하는 제3 기술적 과제는, 상기 제1 기술적 과제에 의한 데이터 압축 방법에 의해 압축된 데이터를 복원하는 하드 카피 장치용 데이터 복원 방법을 제공하는 데 있다.
- <36> 본 발명이 이루고자 하는 제4 기술적 과제는, 상기 하드 카피 장치용 데이터 복원 방법을 수행하는 데이터 복원 장치를 제공하는 데 있다.

【발명의 구성 및 작용】

- <37> 상기 제1 과제를 이루기 위해, 두 가지의 단계들로 촬영된 영상을 하드 카피하기 위해 사용되며, 메모리에 바이트 단위로 저장되는 소스 영상 데이터를 압축하는 본 발명에 의한 하드 카피 장치용 데이터 압축 방법은, 상기 소스 영상 데이터의 열 바이트들을 행 바이트들로 치환하는 (a) 단계 및 현재 압축 처리할 현재 체인의 값과 동일한 값을 갖는 체인이 이전에 압축 처리된 이전 체인들로 이루어진 사전에 존재하는가에 따라 상기 현재 체인과 상기 현재 체인에 후속하는 체인(들)로 이루어진 순차적 체인들 또는 상

기 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 상기 소스 영상 데이터를 압축한 결과로서 결정하는 (b) 단계로 이루어지고, 상기 행 바이트들중 이웃하는 행 바이트들은 이웃하는 메모리 어드레스를 갖고, 상기 열 바이트들중 이웃하는 열 바이트들의 오프셋은 상기 소스 영상 데이터의 행 폭에 해당하고, 상기 압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보 및 헤더에 대한 정보는 상기 (b) 단계가 수행되기 이전에 결정되고 상기 압축한 결과에 포함되며, 상기 체인은 적어도 둘 이상의 연속하는 상기 행 바이트들로 이루어지는 것이 바람직하다.

<38> 상기 제2 과제를 이루기 위해, 두 가지의 단계들로 촬영된 영상을 하드 카피하기 위해 사용되며, 메모리에 바이트 단위로 저장되는 소스 영상 데이터를 압축하는 본 발명에 의한 하드 카피 장치용 데이터 압축 장치는, 입력한 상기 소스 영상 데이터의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 출력하는 제1 데이터 치환부와, 상기 제1 데이터 치환부로부터 출력되는 치환된 결과를 입력하여, 이전에 압축 처리된 이전 체인들중 현재 압축 처리할 현재 체인과 유사하거나 동일한 값을 갖는 체인들로 이루어진 템플릿을 결정하여 출력하는 템플릿 결정부 및 압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보에 응답하여 상기 제1 데이터 치환부로부터 입력한 상기 현재 체인의 값과 동일한 값을 갖는 체인이 상기 템플릿 결정부로부터 입력한 템플릿에 존재하는가를 검사하고, 검사된 결과에 응답하여 상기 현재 체인과 상기 현재 체인에 후속하는 체인(들)로 이루어진 상기 순차적 체인들 또는 상기 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 상기 소스 영상 데이터를 압축한 결과로서 출력하는 엔트로피 엔코더로 구성되고, 상기 행 바이트들중 이웃하는 행 바이트들은 이웃하는 메모리 어드레스를 갖고, 상기 열 바이트들중 이웃하는 열 바이트들의 오프셋은 상기 소스 영상 데이

타의 행 폭에 해당하고, 상기 시작 정보에 대한 정보 및 헤더에 대한 정보는 미리 결정되어 상기 엔트로피 엔코더로 제공되고 상기 압축한 결과에 포함되고, 상기 체인은 적어도 둘 이상의 연속하는 상기 행 바이트들로 이루어지는 것이 바람직하다.

<39> 상기 제3 과제를 이루기 위해, 상기 압축된 소스 영상 데이터로부터 상기 소스 영상 데이터를 복원해내는 본 발명에 의한 하드 카피 장치용 데이터 복원 방법은, 상기 압축된 소스 영상 데이터한 결과로부터 상기 헤더와 상기 시작 정보를 추출하는 (d) 단계와, 상기 압축된 소스 영상 데이터에서 추출한 각 상기 청크 데이터 및 각 상기 체인 데이터에 포함된 압축 체인들을 상기 (d) 단계에서 추출된 상기 헤더 및 상기 시작 정보를 이용하여 엔트로피 디코딩하는 (e) 단계 및 상기 엔트로피 디코딩된 결과의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 복원된 소스 영상 데이터로서 결정하는 (f) 단계로 이루어지는 것이 바람직하다.

<40> 상기 제4 과제를 이루기 위해, 상기 압축된 소스 영상 데이터한 결과로부터 상기 소스 영상 데이터를 복원해내는 본 발명에 의한 하드 카피 장치용 데이터 복원 장치는, 입력한 상기 압축된 소스 영상 데이터로부터 상기 헤더와 상기 시작 정보를 추출하는 정보 추출부와, 상기 정보 추출부로부터 출력되는 상기 헤더 및 상기 시작 정보에 응답하여, 상기 압축된 소스 영상 데이터에서 추출한 각 상기 청크 데이터 및 각 상기 체인 데이터에 포함된 압축 체인들을 엔트로피 디코딩하고, 엔트로피 디코딩된 결과를 출력하는 엔트로피 디코더 및 상기 엔트로피 디코더로부터 출력되는 상기 엔트로피 디코딩된 결과의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 복원된 소스 영상 데이터로서 출력하는 제2 데이터 치환부로 구성되는 것이 바람직하다.

<41> 이하, 본 발명에 의한 하드 카피 장치용 데이터 압축 방법 및 그 방법을 수행하는

데이터 압축 장치의 구성 및 동작을 첨부한 도면들을 참조하여 다음과 같이 설명한다.

- <42> 도 1은 본 발명에 의한 하드 카피 장치용 데이터 압축 방법을 설명하기 위한 플로우차트로서, 해당하는 체인을 인코딩하기 전에 전 처리(pre-processing)하는 단계(제10 ~ 제14 단계들) 및 전 처리된 해당 체인을 엔트로피 엔코딩(entropy encoding)하는 단계(제16 단계)로 이루어진다.
- <43> 도 2는 도 1에 도시된 데이터 압축 방법을 수행하는 본 발명에 의한 데이터 압축 장치의 블럭도로서, 제1 데이터 치환부(20), 템플릿 결정부(22) 및 엔트로피 엔코더(24)로 구성된다.
- <44> 도 3은 치환되지 않은 소스 영상 데이터의 각 바이트가 메모리에 저장된 형태를 나타내는 도면으로서, $p \times q$ 개의 바이트들로 구성된다.
- <45> 도 4는 치환된 소스 영상 데이터의 각 바이트가 메모리에 저장된 형태를 나타내는 도면으로서, $q \times p$ 개의 바이트들로 구성된다.
- <46> 도 2에 도시된 제1 데이터 치환부(20)는 입력단자 IN1을 통해 입력한 도 3에 도시된 소스 영상 데이터의 소정수개의 열 바이트들([1,1], [2,1], ...)(30, 32, ...)을 도 4에 도시된 바와 같이 소정수개의 행 바이트들([1,1], [2,1], ...)(40, 42, ...)로 치환하고, 도 4에 도시된 치환된 결과를 템플릿 결정부(22) 및 엔트로피 엔코더(24)로 각각 출력한다(제10 단계). 여기서, 입력단자 IN1을 통해 입력되는 소스 영상 데이터는 두 가지의 단계들(bilevel)(또는, halftone)로 촬영된(screened) 영상을 하드 카피하기 위해 사용되며, 후술되는 바와 같이 개인용 컴퓨터 파워로부터 발생될 수 있으며, 메모리에 바이트 단위로 저장된다.

<47> 도 3에 도시된 행 바이트들([1,1], [1,2], ... 및 [1,q])(30, 34, ... 및 38)중 이웃하는 행 바이트들([1,1] 및 [1,2])(30 및 34)은 이웃하는 메모리 어드레스를 갖고, 열 바이트들([1,1], [2,1], ... 및 [p,1])(30, 32, ... 및 36)중 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)의 오프셋(offset)은 소스 영상 데이터 비트 맵(bitmap)의 행 폭(row width)으로서 매우 크다. 여기서, 오프셋은 메모리의 거리로서, 오프셋이 크면 어드레싱에 많은 시간이 소요된다.

<48> 제10 단계에서, 소스 영상 데이터를 엔코딩하기 전에 치환하는 이유는 다음과 같은 두 가지의 이유들 때문이다.

<49> 첫 번째 이유는, 두 가지의 단계(halftone-machine)로 영상을 촬영하면 소스 영상을 더 무질서하게 만들기 때문에, 치환된 영상을 무질서화시킬 필요가 없게 된다. 두 번째 이유는, 도 3에 도시된 이웃하는 바이트들의 기하학적 거리는 수평 방향으로 8 픽셀이고 수직 방향으로 1 픽셀이므로 수직 방향으로 이웃하는 바이트들이 수평 방향으로 이웃하는 바이트들보다도 더 비슷하기 때문이다. 예를 들면, 도 3에 도시된 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)은 이웃하는 행 바이트들([1,1] 및 [1,2])(30 및 34)보다도 더 비슷하다. 그럼에도 불구하고, 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)을 어드레싱하는 시간이 이웃하는 행 바이트들([1,1] 및 [1,2])(30 및 34)을 어드레싱하는 시간 보다 길다. 그러므로, 제10 단계를 수행하여, 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)을 도 4에 도시된 바와 같이 수평 방향으로 이웃하는 메모리 어드레스에 위치시킨다. 결국, 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)의 오프셋은 이웃하는 행 바이트들([1,1] 및 [1,2])(30 및 34)의 오프셋보다 크므로, 행 바이트들과 열 바이트들을 치환하여 이웃하는 열 바이트들([1,1] 및 [2,1])(30 및 32)의 오프

셋을 단축시킴으로서, 이웃하는 열 바이트들이 어드레싱되는 시간을 절감시키기 위해, 제10 단계를 수행한다.

<50> 한편, 제10 단계후에, 도 2에 도시된 템플릿 결정부(22)는 제1 데이터 치환부(20)로부터 출력되는 도 4에 도시된 치환된 결과를 입력하여, 이전에 압축 처리된 이전 체인들중 현재 압축 처리할 현재 체인과 유사하거나 동일한 값을 갖는 체인들로 이루어진 템플릿(template)을 결정하고, 결정된 템플릿을 엔트로피 엔코더(24)로 출력한다(제12 단계). 본 발명에서 도입된 '체인'이란, 적어도 둘 이상의 연속하는 바이트들로 이루어진 데이터를 의미하며 이에 대해서는 후술된다. 본 발명의 바람직한 실시예에 의하면, 각 체인은 도 4에 도시된 2개의 행 바이트들([1,1] 및 [2,1])(40 및 42)로 이루어질 수도 있고, 4개의 행 바이트들([1,1], [2,1], ...)(40, 42, ...)로 이루어질 수도 있다. 여기서, 본 발명에 도입된 '템플릿'이란, 현재 체인에 해당하는 체인들을 이전 체인들에서 발견하기 위해 사용되는 서라운드(surround)의 특정 패턴을 의미한다. 이 때, 서라운드란, 현재 체인이 압축 처리되기 이전에 압축 처리된 모든 이전 체인들의 집합을 의미하며 이에 대해서는 후술된다.

<51> 이하, 제12 단계의 템플릿 결정 과정의 본 발명에 의한 바람직한 일실시예를 첨부한 도면들을 참조하여 다음과 같이 설명한다.

<52> 도 5는 메모리에 각 체인이 저장된 형태를 나타내는 도면으로서, 이전에 처리된 이전 체인들(1, 2, 3, ..., n-1 및 n)로 이루어진 그늘진 부분에 해당하는 서라운드 및 현재 체인(c)으로 구성된다.

<53> 도 6 및 도 7들은 길이가 17인 두 가지의 템플릿들의 예시적인 도면들을 나타낸다.

<54> 도 5에 도시된 각 셀은 전술한 체인에 해당하고, 그늘진 부분은 서라운드해 해당한다. 이 때, 이전 체인들(1, 2, 3, ..., n-1 및 n)중 현재 체인(c)과 비슷하거나 동일한 값을 갖는 체인들로 구성된 템플릿은 서라운드로부터 찾는다. 예컨대, 서라운드는 매우 크기 때문에, 압축 시간을 단축시키기 위해 서라운드의 크기를 줄인 템플릿을 구할 필요가 있다. 예를 들면, 도 5에 도시된 서라운드는 도 6 또는 도 7에 도시된 바와 같은 서라운드보다 작은 크기를 갖는 템플릿으로 축소될 수 있다.

<55> 도 8은 본 발명에 의한 템플릿 결정 과정(12)을 설명하기 위한 플로우차트로서, 의사 무작위 체인들 각각에 대해 구한 서라운드를 이용하여 템플릿을 결정하는 단계(제50 ~ 제56 단계)로 이루어진다.

<56> 도 9는 메모리(60)에 바이트 단위로 저장된 소스 영상 데이터를 체인 형태로 나타내는 도면으로서, 각 셀은 체인을 나타내고, 그늘진 부분들은 서라운드들(68, 70 및 72)을 나타낸다.

<57> 도 10은 도 8에 도시된 템플릿 결정 과정(12)에 의해서 결정된 템플릿의 예시적인 도면으로서, 그늘진 부분은 현재 체인(c)에 대해 m개의 체인들(1, 2, ..., m-2, m-1 및 m)로 이루어지는 예시적인 템플릿을 나타낸다.

<58> 템플릿을 구하기 위해, 템플릿 결정부(22)는 제10 단계에서 치환된 도 9에 도시된 메모리(60)에 저장된 소스 영상 데이터에 적용한 의사 무작위 체인(pseudo-random chain)들(62, 64 및 66)에 대해 가능한 서라운드들(68, 70 및 72)을 구한다(제50 단계). 예컨대, 도 10에 도시된 현재 체인(c)의 템플릿을 구하기 위해, 도 9에 도시된 바와 같이 몇 개의 의사 무작위 체인들(62, 64 및 66)을 테스트한다. 여기서, 각 서라운드(68, 70 또는 72)는 의사 무작위 체인(62, 64 또는 66)이 처리되기 이전에 처리된 이전 체인

들(1, 2, ..., n-1 및 n)로 이루어지며, 크기 n을 갖는다.

<59> 제50 단계후에, 각각의 서라운드(68, 70 또는 72)에서 해당하는 의사 무작위 체인(62, 64 또는 66)의 값과 동일한 값을 갖는 체인(들)을 카운팅한다(제52 단계). 예컨데, 서라운드(68)에 포함된 체인들(1, 2, ..., n-1 및 n)중 의사 무작위 체인(c)(62)의 값과 동일한 값을 갖는 체인을 카운팅하고, 서라운드(70)에 포함된 체인들(1, 2, ..., n-1 및 n)중 의사 무작위 체인(c)(64)의 값과 동일한 값을 갖는 체인을 카운팅하고, 서라운드(72)에 포함된 체인들(1, 2, ..., n-1 및 n)중 의사 무작위 체인(c)(66)의 값과 동일한 값을 갖는 체인을 카운팅한다. 이와 같이, 모든 의사 무작위 체인들(62, 64 및 66)에 대해서 이와 같은 카운팅 동작을 반복한다.

<60> 제52 단계후에, 각 서라운드(68, 70 또는 72)에 포함된 체인들(1, 2, ..., n-1 및 n) 각각을 표시하는 인덱스 예를 들면 '-1', '-2', ..., '-62' 및 '-63'별로 제52 단계에서 카운팅된 값을 분류한다(제54 단계).

<61> 제54 단계후에, 분류된 카운팅된 값들중 가장 큰 값부터 소정 개수($m \leq n$)(여기서, m은 템플릿의 길이를 나타내며 미리 정해진다.)의 카운팅된 값들을 선택한다(제56 단계). 즉, 모든 의사 무작위 체인들(62, 64 및 66)에 대해서 더 많은 동일한 값을 제공하는 인덱스들만을 선택한다. 예를 들면, 도 10에 도시된 바와 같이, 제56 단계에서 선택된 카운팅된 값들에 해당하는 인덱스들이 표시하는 체인들(1, 2, ..., m-2, m-1 및 m)로 이루어지는 템플릿($T[1, 2, 3, \dots \text{ 및 } m]$)이 최종적으로 결정될 수 있다. 여기서, $T[]$ 이란, m개 정수들의 배열을 의미하고, $[]$ 의 내부에 있는 각 정수(1, 2, 3, ... 또는 m)는 오프셋을 의미한다. 예를 들어, c가 현재 체인의 인덱스라고 할 때, 템플릿에 존재하는 어느 $i(1 \leq i \leq m)$ 가 표시하는 체인의 인덱스(I)는 다음 수학적 식 1과 같이 계산될

수 있다.

<62> 【수학식 1】

$$I = c - T[i]$$

<63> 여기서, $T[i]$ 는 템플릿에 포함되는 어느 i 가 표시하는 체인과 현재 체인(c) 사이의 거리를 나타낸다. 이 때, 치환된 소스 영상 데이터(60)의 각 행 데이터에서 가장 처음에 어드레싱되는 체인은 그 행 데이터의 이전 행 데이터에서 가장 나중에 어드레싱되는 체인의 서라운드에 속하는 경우도 있다.

<64> 전술한 바와 같이 템플릿을 결정할 때, 도 9에 도시된 메모리(60)에 저장된 치환된 소스 영상 데이터에 단지 3개의 의사 무작위 체인들(62, 64 및 66)만이 적용되었으나, 의사 무작위 체인의 수는 3개로 국한되지 않고 더 많은 수로 확대될 수 있다.

<65> 한편, 제12 단계후에, 압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보(k) 및 헤더에 대한 정보를 결정한다(제14 단계).

<66> 여기서, 시작 정보를 결정하는 방법은 다음과 같이 두 가지가 있다.

<67> 첫 번째, 시작 정보(k)는 템플릿에 포함되는 체인들중 압축 처리를 시작할 체인에 대한 정보를 갖는 것으로 결정될 수 있으며, 이 때, 압축 처리를 시작할 체인은 히스토리(history)를 갖어야 한다. 여기서, 히스토리란, 시작 정보(k)가 표시하는 체인보다 이전에 압축 처리된 이전 체인(들)의 배열을 의미한다. 예컨대, 도 9에 도시된 치환된 소스 영상 데이터에서 체인(80)($k=0$)부터 체인(82)까지 오른쪽 방향 및 아래 방향으로 체인들에 대한 압축 처리를 진행한다고 하자. 이 때, 체인(80)은 히스토리를 갖지 않으므로 시작 정보(k)가 표시하는 체인이 될 수 없다.

즉, 시작 정보(k)는 0이 될 수 없다. 따라서, 시작 정보(k)는 1이상의 값이 된다. 예컨데, 압축 처리는 치환된 소스 영상 데이터에서 첫번째 체인(80)을 제외한 체인들중 어느 체인부터 수행될 수 있다. 이 때, 시작 정보(k)가 '1'인 경우, 본 발명에 의한 데이터 압축 방법에 의하면, 제14 단계에서, 템플릿에 포함되는 어느 i번째 체인이 유효한 체인인가를 판단하여, 그 i번째 체인이 유효한 체인이 아니면, i번째 체인은 그 템플릿에서 제거된다. 여기서, 어느 i번째 체인의 수학적 1에 표현된 인덱스(I)가 음이 아닌 경우에 그 i번째 체인은 유효한 체인으로 결정된다.

<68> 두 번째, 시작 정보(k)는 템플릿에 포함되는 체인들중 가장 큰 $T[i](\max T[i])$ 를 갖는 체인에 대한 정보를 갖는 것으로 결정될 수 있다. 이 경우, 제14 단계는 전술한 바와 같이 어느 i번째 체인이 유효한가를 판단하지 않아도 된다.

<69> 제14 단계후에, 도 2에 도시된 엔트로피 엔코더(24)는 입력단자 IN2를 통해 입력한 시작 정보(k)에 응답하여, 제1 데이터 치환부(20)로부터 현재 체인(c) 및 사전(dictionary)을 입력하고, 현재 체인(c)의 값과 동일한 값을 갖는 체인이 사전에 존재하는가를 검사하고, 검사된 결과에 응답하여 순차적(sequential) 체인들 또는 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 합성하여 생성한 압축된 소스 영상 데이터를 출력단자 OUT1을 통해 출력한다(제16 단계). 여기서, 순차적 체인들이란, 현재 체인과 그 현재 체인에 후속하는 체인(들)을 일컫는다. 일반적인 엔트로피 엔코딩 방식에는 사전 방식 및 호프만 방식(Hoffman method)이 있고, 전술한 본 발명에 의한 엔트로피 엔코딩 방식은 사전 방식에 의해 수행된다.

<70> 여기서, 사전이란, 현재 체인이 압축 처리되기 이전에 압축 처리된 체인인 이전 체인들로 이루어져 있으며, 전술한 서라운드 또는 템플릿에 해당한다. 만일, 사전이 템플

릿인 경우, 본 발명에 의한 데이터 압축 방법은 도 1에 도시된 바와 같고, 본 발명에 의한 데이터 압축 장치는 도 2에 도시된 바와 같이 구성된다. 그러나, 사전이 서라운드인 경우, 도 1에 도시된 본 발명에 의한 데이터 압축 방법에서 템플릿을 결정하는 단계(제 12 단계)는 생략되므로 제10 단계후에 제14 단계가 수행되고, 도 2에 도시된 본 발명에 의한 데이터 압축 장치에서 템플릿 결정부(22)는 서라운드 결정부(미도시)로 대체된다. 물론, 서라운드 결정부(미도시)는 현재 체인의 서라운드를 결정한다.

<71> 이하, 본 발명의 이해를 돕기 위해 사전이 템플릿이라 가정하고 본 발명에 의한 데이터 압축 및 복원 방법들과 그 장치들을 설명한다.

<72> 전술한 가정하에서, 엔트로피 엔코더(24)는 템플릿 결정부(22)로부터 입력한 템플릿에 제1 데이터 치환부(20)로부터 입력한 현재 체인이 존재하는가를 검사하고, 검사된 결과에 응답하여 해당하는 체인들을 엔트로피 엔코딩한 결과를 출력단자 OUT1을 통해 출력한다.

<73> 도 2에 도시된 엔트로피 엔코더(24)에서 수행되는 본 발명에 의한 엔트로피 엔코딩 과정(16)의 본 발명에 의한 바람직한 일실시예에 대해 첨부한 도면들을 참조하여 다음과 같이 설명한다.

<74> 도 11은 본 발명에 의한 엔트로피 엔코딩 과정(16)을 설명하기 위한 플로우차트로서, 현재 체인이 매칭 체인을 갖는가에 따라 현재 체인을 달리 엔코딩하는 단계(제90 ~ 제104 단계들)로 이루어진다.

<75> 도 12는 도 2에 도시된 엔트로피 엔코더(24)의 본 발명에 의한 바람직한 일실시예의 블록도로서, 데이터 입력부(110), 제1 매칭 검사부(112), 제1 카운터(114), 제1 부호

화부(116), 제2 카운터(118), 비교부(120), 제2 부호화부(122), 저장부(124), 제1 제어부(126) 및 제1 데이터 출력부(128)로 구성된다.

<76> 도 12에 도시된 데이터 입력부(110)는 입력단자 IN2를 통해 입력한 시작 정보와 헤더에 대한 정보들 및 제1 제어부(126)로부터 입력한 제1 제어 신호(C1)에 응답하여 제1 데이터 치환부(20)로부터 입력단자 IN3을 통해 입력한 현재 체인의 값을 읽고, 읽혀진 현재 체인의 값을 제1 매칭 검사부(112)로 출력한다(제90 단계).

<77> 제90 단계후에, 제1 매칭 검사부(112)는 데이터 입력부(110)에서 읽혀진 현재 체인의 값과 동일한 값을 갖는 체인인 매칭 체인(matching chain)이 템플릿 결정부(22)로부터 입력단자 IN4를 통해 입력한 템플릿에 존재하는가를 판단한다(제92 단계). 이 때, 제1 매칭 검사부(112)는 판단된 결과로부터 매칭 체인의 유무를 나타내는 제1 매칭 신호 및 검출될 수 있는 매칭 체인의 i를 검출하고, 검출된 제1 매칭 신호를 저장부(124), 제1 및 제2 카운터들(114 및 118)로 출력하고, 검출된 i를 제1 부호화부(116)로 출력한다.

<78> 만일, 매칭 체인이 템플릿에 존재하면, 매칭 체인의 i 및 매칭 체인을 갖는 체인들이 순차되는 개수를 카운팅한 결과인 제1 카운팅 값(count1)을 부호화한다(제94 단계). 이를 위해, 제1 카운터(114)는 제1 매칭 검사부(112)로부터 출력되는 제1 매칭 신호를 통해 매칭 체인이 템플릿에 존재하는 것으로 인식되면, 매칭 체인을 갖는 체인들이 순차되는 개수를 카운팅하고, 카운팅된 결과인 제1 카운팅 값(count1)을 제1 부호화부(116)로 출력한다. 이 때, 제1 부호화부(116)는 제1 카운팅 값(count1) 및 매칭 체인의 i를 부호화하고, 부호화된 결과를 저장부(124)로 출력한다. 이 때, 제1 카운터(114)는 서로 동일한 값을 갖는 체인들이 순차되는 개수를 카운팅하여 제1 카운팅 값(count1)을 구하

고, 제1 카운팅 값을 런 길이 부호화(RLC:Run Length Code) 방식에 의해 부호화할 수도 있다. RLC에는 반복되는 체인의 수와 이러한 반복이 시작되는 체인에 대한 정보만을 부호화한다.

<79> 그러나, 매칭 체인이 템플릿에 존재하지 않은 것으로 판단되면, 매칭 체인을 갖지 못하는 체인들이 순차되는 개수를 카운팅하여 제2 카운팅 값(count2)을 구하고, 현재 체인의 값을 저장한다(제96 단계). 이를 위해, 제2 카운터(118)는 제1 매칭 신호를 통해 매칭 체인이 템플릿에 존재하지 않은 것으로 인식되면, 매칭 체인을 갖지 못하는 체인들이 순차되는 개수를 카운팅하고, 카운팅된 결과를 제2 카운팅 값(count2)으로서 비교부(120) 및 제2 부호화부(122)로 출력한다. 또한, 입력단자 IN3을 통해 입력된 현재 체인은 저장부(124)로 출력된다.

<80> 한편, 제94 단계 또는 제96 단계후에, 제1 제어부(126)는 입력단자 IN3을 통해 제1 데이터 치환부(20)로부터 입력한 현재 체인이 치환된 소스 영상 데이터에서 마지막으로 처리할 체인인가 예를 들면 도 9에 도시된 체인(82)인가를 판단하고, 판단된 결과를 제1 제어 신호(C1)로서 데이터 입력부(110) 및 비교부(120)로 각각 출력한다(제98 단계).

<81> 만일, 현재 체인이 마지막으로 처리할 체인이 아니면, 현재 체인에 후속하는 체인을 새로운 현재 체인으로 결정하고, 제90 단계로 진행한다(제100 단계). 이를 위해, 제1 제어부(126)로부터 출력되는 제1 제어 신호(C1)를 통해, 입력단자 IN3을 통해 입력한 현재 체인이 마지막 체인이 아닌 것으로 인식되면, 데이터 입력부(110)는 현재 체인에 후속하는 다음 체인을 새로운 현재 체인으로서 입력단자 IN3을 통해 새롭게 입력한다.

<82> 그러나, 현재 체인이 마지막으로 처리할 체인이면, 제2 카운팅 값(count2)에 따라 소정 초기값과 제2 카운팅 값을 부호화한다(제102 단계). 이를 위해, 비교부(120)는 제1

제어부(126)로부터 출력되는 제1 제어 신호(C1)를 통해 현재 체인이 마지막 체인인 것으로 인식되면, 제2 카운팅 값(count2)을 소정 임계값(th) 예를 들면 '0'과 비교하고, 비교된 결과를 제2 부호화부(122)로 출력한다. 제2 부호화부(122)는 비교부(120)에서 비교된 결과에 응답하여, 제2 카운팅 값(count2)과 외부로부터 입력한 소정 초기값(In)을 부호화하고, 부호화된 결과를 저장부(124)로 출력한다.

<83> 이 때, 저장부(124)는 제1 매칭 검사부(112)로부터 출력되는 제1 매칭 신호에 응답하여, 제1 및 제2 부호화부들(116 및 122)에서 부호화된 결과들, 제1 데이터 치환부(20)로부터 입력단자 IN3을 통해 입력한 현재 체인의 값 및 입력단자 IN2를 통해 입력한 시작 정보(k)를 저장한다. 이를 위해, 저장부(124)는 제1 및 제2 저장부들(미도시)로 구성될 수 있다. 제1 저장부(미도시)는 제1 매칭 신호에 응답하여, 제1 부호화부(116)에서 부호화된 결과 또는 제2 부호화부(122)에서 부호화된 결과를 후술되는 청크 데이터(chunk data)로서 저장한다. 제2 저장부(미도시)는 제1 매칭 신호에 응답하여, 현재 체인의 값 및 시작 정보(k)를 후술되는 체인 데이터(chain data)로서 저장한다.

<84> 제102 단계후에, 제1 데이터 출력부(128)는 저장부(124)로부터 입력한 시작 정보, 현재 체인의 값, 제1 및 제2 부호화부들(116 및 122)에서 부호화된 결과들을 합성하고, 합성된 결과를 압축된 소스 영상 데이터로서 출력단자 OUT1을 통해 출력한다(제104 단계).

<85> 이하, 도 1 또는 도 2에 도시된 본 발명에 의한 하드 카피 장치용 데이터 압축 방법 및 장치에 의해 압축된 소스 영상 데이터의 포맷을 첨부한 도면들을 참조하여 다음과 같이 살펴본다.

<86> 도 13은 압축한 소스 영상 데이터에 대한 본 발명에 의한 바람직한 일실시예의 포

맷을 나타내는 도면으로서, 파일 헤더(file header)(140), 체인 데이터(142) 및 청크 데이터(144)로 구성된다.

<87> 일반적으로 대부분의 컴퓨터들은 어느 데이터들이 4바이트들로 정렬되어 있으면 32비트에서 보다 빠르게 동작하고, 어느 데이터들이 16비트들로 정렬되어 있으면 16비트에서 보다 빠르게 동작하다. 그러므로, 본 발명에 의해 압축된 소스 영상 데이터는 도 13에 도시된 바와 같이 청크 데이터 및 체인 데이터들로 별도로 구성되었다. 청크 데이터(144)는 1바이트로 정렬된 데이터이지만, 체인 데이터(142)는 4 바이트들 또는 2 바이트들이 될 수 있다. 이 때, 체인 데이터만이 정렬된다.

<88> 도 13에 도시된 청크 데이터(144)는 전술한 바와 같이 제1 및 제2 부호화부들(116 및 122)에서 부호화된 결과들을 포함하고 있으며, 단위 바이트로 정렬된(one-byte aligned) 데이터에 해당하며, 엔트로피 엔코딩이 종료한 후에 제104 단계에서 통합(merge)된다. 여기서, 단위 바이트로 정렬된 데이터란, 데이터가 정렬되지 않았음을 의미한다. 왜냐하면, 청크 데이터(144)는 서로 다른 크기를 갖는 데이터들로 이루어지기 때문에 정렬되지 않는다. 예컨대, 청크 데이터(144)는 정렬될 때도 있고 정렬되지 않을 때도 있는 것이 아니라 항상 정렬되지 않는다.

<89> 도 14는 도 13에 도시된 파일 헤더(140)에 대한 본 발명에 의한 바람직한 일실시예의 데이터 포맷을 나타내는 도면으로서, 매직 헤더(magic header)(150), 플래그(flag)(152), 체인 데이터의 크기(154), 템플릿의 길이(m)(156), 템플릿(T[1, 2, ..., m])(158), 예비 공간(reserved space)(160) 및 상보형 더미 바이트들(complementary dummy bytes)(162)로 구성된다.

<90> 도 13에 도시된 파일 헤더(140)는 고정된 포맷이며, 제14 단계에서 언급한 헤더에

해당하고, 도 14에 도시된 바와 같이 구현될 수 있다. 도 14에 도시된 파일 헤더(140)에서 체인 데이터의 크기(154)는 본 발명에 의한 데이터 압축 장치 및 데이터 복원 장치를 포함하는 시스템의 특성에 의존하기 때문에, 성능 향상을 위해서 파일 헤더(140)에 이 값(154)을 유지시키지 않는 것이 매우 중요하다. 특히, 데이터 압축 및 복원 장치들을 설계할 때, 체인 데이터의 크기(154)에 대하여 고려해주어야 한다. 예컨대, 소스 영상 데이터의 전체 크기와 체인 데이터의 크기는 정확하게 동일하여야 한다. 또한, 도 14에 도시된 예비 공간(160)은 영상의 폭과 높이를 포함할 수도 있다. 그러나, 영상의 폭과 높이인 크기는 종종 미리 정해지기 때문에, 데이터 압축 장치의 설계시 필수적으로 고려할 사항은 아니다.

<91> 도 15는 도 13에 도시된 체인 데이터(142)에 대한 본 발명에 의한 바람직한 일실시예의 포맷을 나타내는 도면으로서, 시작 정보(k)(170) 및 매칭 체인을 갖지 못하는 현재 체인의 값(172)으로 구성된다.

<92> 도 13에 도시된 체인 데이터(142)의 시작 점(origin)은 파일 헤더(140)의 끝점(end) 다음에 위치하고 도 15에 도시된 바와 같이 구현될 수 있다. 도 15에 도시된 체인 데이터(142)에서 시작 정보(k)(170)는 전술한 바와 같이 압축 처리를 시작할 체인에 대한 정보를 가지고 있다. 이 때, 압축 처리가 시작되는 체인에 대한 정보(k)는 부호화되지 않고 시작 정보(170)에 그대로 저장된다. 이 때, 전술한 제96 단계에서 설명한 바와 같이, 매칭 체인을 갖지 못하는 현재 체인에 대한 정보는 도 15에 도시된 소정 부분(172)에 부호화되지 않은 채 저장된다.

<93> 도 16a 및 도 16b들은 도 13에 도시된 청크 데이터(144)에 대한 본 발명에 의한 포맷을 나타내는 도면으로서, 도 16a에 도시된 데이터를 제1 바이트(1st byte)라 하고, 도

16b에 도시된 데이터를 제2 바이트(2nd byte)라 하자. 이 때, 제2 바이트(2nd byte)는 선택적으로 마련된다.

<94> 전술한 바와 같이, 매칭 체인의 i(또는 소정 초기값)와 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 16a에 도시된 제1 바이트(1st byte)에 함께 저장될 수 있다. 이 때, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 16a에 도시된 제1 바이트(1st byte) 뿐만 아니라 도 16b에 도시된 제2 바이트(2nd byte)에도 저장될 수 있다. 이 때, 매칭 체인의 i(또는 소정 초기값)는 $M(m=2^M-1)$ 비트로서 항상 도 16a에 도시된 제1 바이트(1st byte)에 저장되고, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 제1 바이트(1st byte)의 나머지 비트들인 $8-M$ 에 저장된다. 여기서, 소정 초기값은 제1 바이트(1st byte)에 M개의 예를 들면 0비트들로서 저장될 수 있다. 만일, 제1 바이트(1st byte)에 저장된 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 '0'이면, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 '0'이 될 수 없기 때문에 실질적으로 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 $8-M$ 비트들에서 저장할 수 있는 최대 가능한 값($2^{8-M}-1$)을 초과할 경우이며, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 제1 바이트(1st byte)의 $8-M$ 비트들 뿐만 아니라 제2 바이트(2nd byte)도 사용해서 저장된다. 여기서, 본 발명에 의하면, M은 3, 4, 5 또는 $6(m=7, 15, 31 \text{ 또는 } 63)$ 으로 될 수 있다.

<95> 이하, m의 값에 따라 체크 데이터(144)가 데이터를 달리 저장하는 방법에 대해 첨부한 도면들을 참조하여 다음과 같이 설명한다.

<96> 도 17은 $m=7$ 인 경우, 즉, 1 바이트로만 된 체크 데이터(144)의 포맷을 나타내는 도면이다.

- <97> 0 ~ 7중에서 어느 값을 가지는 매칭 체인의 i 또는 예를 들면 $i=0$ 으로 설정될 수 있는 소정 초기값은 도 17에 도시된 3개의 비트들(0, 1 및 2)에 저장되고, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 17에 도시된 5개의 비트들(3, 4, 5, 6 및 7)에 저장된다.
- <98> 도 18a 및 18b들은 $m=15$ 인 경우, 즉, 1 또는 2 바이트(들)로 된 체크 데이터(144)의 포맷을 나타내는 도면들이다.
- <99> 0 ~ 15중에서 어느 값을 가지는 매칭 체인의 i 또는 예를 들면 $i=0$ 으로 설정될 수 있는 소정 초기값은 도 18a에 도시된 4개의 비트들(0, 1, 2 및 3)에 저장되고, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 18a에 도시된 4개의 비트들(4, 5, 6 및 7)에 저장된다. 이 때, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 15를 초과할 경우, 도 18b에 도시된 제2 바이트에 저장된다. 그러므로, 최대로 저장될 수 있는 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 271이 되고, 272 종류(0 ~ 271)의 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 저장될 수 있다.
- <100> 도 19a 및 19b들은 $m=31$ 인 경우, 즉, 1 또는 2 바이트(들)로 된 체크 데이터(144)의 포맷을 나타내는 도면들이다.
- <101> 0 ~ 31중에서 어느 값을 가지는 매칭 체인의 i 또는 예를 들면 $i=0$ 으로 설정될 수 있는 소정 초기값은 도 19a에 도시된 5개의 비트들(0, 1, 2, 3 및 4)에 저장되고, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 19a에 도시된 3개의 비트들(5, 6 및 7)에 저장된다. 이 때, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 7을 초과할 경우, 도 19b에 도시된 제2 바이트에 저장된다. 그러므로, 최대로 저장될 수 있는 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 263이 되고, 264 종류(0 ~

263)의 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 저장될 수 있다.

<102> 도 20a 및 20b들은 $m=63$ 인 경우, 즉, 1 또는 2 바이트(들)로 된 체크 데이터(144)의 포맷을 나타내는 도면들이다.

<103> 0 ~ 63중에서 어느 값을 가지는 매칭 체인의 i 또는 예를 들면 $i=0$ 으로 설정될 수 있는 소정 초기값은 도 20a에 도시된 6개의 비트들(0, 1, 2, 3, 4 및 5)에 저장되고, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 도 20a에 도시된 2개의 비트들(6 및 7)에 저장된다. 이 때, 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 3을 초과할 경우, 도 20b에 도시된 제2 바이트에 저장된다. 그러므로, 최대로 저장될 수 있는 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]은 259가 되고, 260 종류(0 ~ 259)의 제1 카운팅 값(count1)[또는 제2 카운팅 값(count2)]이 저장될 수 있다.

<104> 이하, 도 11에 도시된 제94, 제96 및 제102 단계들 각각의 본 발명에 의한 바람직한 일실시예를 다음과 같이 살펴본다. 이 때, 다음에 설명되는 실시예들은 도 12에 도시된 데이터 압축 장치에서 반드시 수행되어야 하는 것을 아니며, 이를 제외하면, 전술한 설명들은 이러한 실시예들에 모두 그대로 적용된다. 예컨대, 도 12에 도시된 데이터 압축 장치는 제1 및 제2 카운터들(114 및 118)을 갖지만, 다음에 설명되는 실시예들에서는 하나의 카운터만을 사용할 수도 있다.

<105> 도 21은 도 11에 도시된 제94 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트로서, 제2 카운팅 값(count2)을 플러쉬(flush)하고 매칭 체인의 i 와 제1 카운팅 값(count1)을 부호화하는 단계(제180 및 제182 단계들)로 이루어진다.

<106> 먼저, 제92 단계에서 매칭 체인이 템플릿에 존재하는 것으로 판단되면, 제2 카운팅

값(count2)을 플러쉬한다(제180 단계). 제180 단계후에, 매칭 체인의 i 및 제1 카운팅 값을 부호화하고, 제98 단계로 진행한다(제182 단계).

<107> 도 22는 도 11에 도시된 제96 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트로서, 현재 체인의 값을 저장하고 제2 카운팅 값(count2)을 구하는 단계(제190 ~ 제196 단계들)로 이루어진다.

<108> 먼저, 제92 단계에서 매칭 체인이 템플릿에 존재하지 않는 것으로 판단되면, 현재 체인의 값을 도 13에 도시된 체인 데이터(142)로 결정하여 저장한다(제190 단계). 제190 단계후에, 제2 카운팅 값(count2)을 1만큼 증가시킨다(제192 단계). 이는, 전술한 바와 같이 매칭되지 않은 현재 체인의 개수를 카운팅하기 위함이다. 제192 단계후에, 제2 카운팅 값(count2)이 소정 최대값인가를 판단하고, 제2 카운팅 값(count2)이 소정 최대값이 아니면, 제98 단계로 진행한다(제194 단계). 그러나, 제2 카운팅 값(count2)이 소정 최대값이면, 제2 카운팅 값(count2)을 플러쉬하고 제98 단계로 진행한다(제196 단계).

<109> 도 23은 도 11에 도시된 제102 단계의 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트로서, 제2 카운팅 값(count2)에 따라 소정 초기값과 제2 카운팅 값(count2)을 부호화하는 단계(제200 및 제202 단계들)로 이루어진다.

<110> 먼저, 제98 단계에서 현재 체인이 마지막으로 처리할 체인인 것으로 판단되면, 제2 카운팅 값(count2)이 소정 임계값(th) 예를 들면 0보다 큰가를 판단하고, 제2 카운팅 값(count2)이 0보다 크지 않으면 제104 단계로 진행한다(제200 단계). 그러나, 제2 카운팅 값(count2)이 0보다 크면, 소정 초기값과 제2 카운팅 값(count2)을 부호화하고, 제104 단계로 진행한다(제202 단계).

<111> 도 24는 메모리에 저장된 데이터의 형태를 도면으로서, 그늘진 부분은 예시적인 템플릿에 해당하고, c는 현재 체인을 나타낸다.

<112> 도 25a는 메모리에 저장된 데이터의 형태를 도면으로서, 도 24에 도시된 각 체인이 가질 수 있는 예시적인 값을 나타내며, 그늘진 부분은 도 24에 도시된 템플릿에 해당한다.

<113> 도 25b는 메모리에 저장된 데이터의 형태를 도면으로서, 도 24에 도시된 각 체인이 가질 수 있는 다른 예시적인 값을 나타내며, 그늘진 부분은 도 24에 도시된 템플릿에 해당한다.

<114> 도 25c는 메모리에 저장된 데이터의 형태를 도면으로서, 도 24에 도시된 각 체인이 가질 수 있는 또 다른 예시적인 값을 나타내며, 그늘진 부분은 도 24에 도시된 템플릿에 해당한다.

<115> 이하, 본 발명의 이해를 돕기 위해, 도 1에 도시된 제12 단계를 수행하여 도 24에 도시된 바와 같은 템플릿이 구해진다고 가정하고, 도 24에 도시된 템플릿의 각 체인(1, 2, 3, ..., 16 및 17)이 가질 수 있는 값(A, B, C, X 또는 Y)이 도 25a, 25b 또는 25c와 같이 된다고 가정할 때, 전술한 도면들을 참조하여 도 1에 도시된 제16 단계인 본 발명에 의한 엔트로피 엔코딩을 다음과 같이 설명한다. 여기서, A, B, C, X 또는 Y는 서로 다른 데이터이다.

<116> 첫 번째로, 각 체인의 값이 도 25a와 같다고 가정하자.

<117> 먼저, 현재 압축 처리할 도 25a에 도시된 현재 체인(c)(210)의 값(A)을 읽는다. 이 때, 현재 체인(c)(210)의 값(A)과 동일한 값을 갖는 매칭 체인이 도 25a에 도시된 템플

릿의 내부에 존재하는가를 판단한다. 여기서, 매칭 체인이 존재하므로, 템플릿 내부에 존재하는 매칭 체인들($i=1, 8, 10$ 및 13)중 최대로 순차되는 순차적 체인들($210, 212$ 및 214)의 값들(A, B 및 C)과 동일한 값들(A, B 및 C)을 갖는 순차적인 체인들($220, 222$ 및 224)이 시작되는 매칭 체인(220)의 i 인 '13'을 찾는다. 따라서, 매칭 체인의 i 인 '13'과 매칭 체인을 갖는 체인들($210, 212$ 및 214)이 순차되는 개수인 $\text{count1}='3'$ 을 체인 데이터로서 부호화한다.

<118> 두 번째로, 각 체인의 값이 도 25b와 같다고 가정하자.

<119> 먼저, 현재 압축 처리할 도 25b에 도시된 현재 체인(230)의 값(A)을 읽는다. 이 때, 현재 체인(230)의 값(A)과 동일한 값을 갖는 매칭 체인이 도 25b에 도시된 템플릿의 내부에 존재하는가를 판단한다. 여기서, 매칭 체인(236)이 존재하므로, 템플릿 내부에 존재하는 매칭 체인(236)의 i 인 1을 찾는다. 또한, 매칭 체인(236)을 갖는 순차적 체인들($230, 232$ 및 234)의 개수를 카운팅하여 제1 카운팅 값($\text{count1}=3$)을 구한다. 따라서, 매칭 체인의 i 인 1과 제1 카운팅 값(count1)인 3을 체인 데이터로서 부호화한다. 예컨대, 도 25b에 도시된 서로 같은 값을 갖는 체인들($236, 230, 232$ 및 234)은 전술한 바와 같이 RLC 방식에 의해서 부호화된다.

<120> 세 번째로, 각 체인의 값이 도 25c와 같다고 가정하자.

<121> 먼저, 현재 압축 처리할 도 25c에 도시된 현재 체인(c)(240)의 값(A)을 읽는다. 이 때, 현재 체인(c)(240)의 값(A)과 동일한 값을 갖는 매칭 체인이 도 25c에 도시된 템플릿의 내부에 존재하는가를 판단한다. 여기서, 매칭 체인이 존재하지 않으므로, 매칭 체인을 갖는 못하는 체인들($240, 242$ 및 244)이 순차되는 개수인 제2 카운팅 값($\text{count2}=3$)을 부호화하여 청크 데이터로서 저장하고, 매칭 체인을 갖지 못하는 순차적 체인들($240,$

242 및 244)의 값들(A, B 및 C)을 압축하지 않고 체인 데이터로서 저장한다. 이 때, 도 16a, 도 17, 도 18a, 도 19a 또는 도 20a에 도시된 청크 데이터의 제1 바이트에서 i가 저장되는 위치에 소정 초기값이 0으로 저장될 수 있다.

<122> 이하, 본 발명에 의한 하드 카피 장치용 데이터 복원 방법 및 그 방법을 수행하는 장치의 구성 및 동작을 첨부한 도면들을 참조하여 다음과 같이 설명한다.

<123> 도 26은 본 발명에 의한 하드 카피 장치용 데이터 복원 방법을 설명하기 위한 플로 우차트로서, 압축된 소스 영상 데이터로부터 추출한 헤더와 시작 정보를 이용하여 추출한 청크 및 체인 데이터들에 포함된 압축 체인을 엔트로피 디코딩하는 단계(제300 및 제302 단계들) 및 엔트로피 디코딩된 결과를 치환하여 소스 영상 데이터를 복원하는 단계(제304 단계)로 이루어진다.

<124> 도 27은 도 26에 도시된 데이터 복원 방법을 수행하는 본 발명에 의한 데이터 복원 장치의 블럭도로서, 정보 추출부(310), 엔트로피 디코더(312) 및 제2 데이터 치환부(314)로 구성된다.

<125> 도 26 및 도 27에 도시된 방법 및 장치는 전술한 본 발명에 의한 데이터 압축 방법 및 장치에 의해서 압축된 소스 영상 데이터를 원래의 소스 영상 데이터로 복원해낸다.

<126> 이를 위해, 먼저, 정보 추출부(310)는 입력단자 IN5를 입력한 압축된 소스 영상 데이터로부터 도 13에 도시된 헤더(140)와 도 15에 도시된 시작 정보(170)를 추출하고, 추출된 헤더(140)와 시작 정보(170)를 엔트로피 디코더(312)로 출력한다(제300 단계).

<127> 제300 단계후에, 엔트로피 디코더(312)는 입력단자 IN5를 통해 입력한 압축된 소스 영상 데이터로부터 헤더(140) 및 시작 정보(170)에 응답하여 추출한 도 13에 도시된 청

크 데이터(144) 및 체인 데이터(142)에 포함된 압축된 체인인 압축 체인들을 엔트로피 디코딩하고, 엔트로피 디코딩된 결과를 제2 데이터 치환부(314)로 출력한다(제302 단계). 예컨대, 제302 단계는 도 11에 도시된 엔트로피 엔코딩 과정(16)을 역으로 수행한다.

<128> 이하, 전술한 본 발명에 의한 엔트로피 디코딩 과정(302)과 그 과정을 수행하는 본 발명에 의한 엔트로피 디코더(312)의 구성 및 동작을 첨부한 도면들을 참조하여 다음과 같이 설명한다.

<129> 도 28은 도 26에 도시된 제302 단계를 수행하는 본 발명에 의한 바람직한 일실시예를 설명하기 위한 플로우차트로서, 추출한 모든 청크 데이터들 각각이 갖는 값에 따라서 압축 체인을 달리 복원하여 엔트로피 디코딩된 결과를 구하는 단계(제320 ~ 제330 단계들)로 이루어진다.

<130> 도 29는 도 27에 도시된 엔트로피 엔코더(312)의 본 발명에 의한 바람직한 일실시예의 블럭도로서, 데이터 추출부(340), 제2 매칭 검사부(342), 제1 및 제2 체인 복원부들(344 및 346), 제2 데이터 출력부(348) 및 제2 제어부(350)로 구성된다.

<131> 먼저, 데이터 추출부(340)는 정보 추출부(310)로부터 입력단자 IN6을 통해 입력한 헤더(140) 및 시작 정보(170)와 제2 제어부(350)로부터 입력한 제2 제어 신호(C2)에 응답하여, 입력단자 IN5를 통해 입력한 압축된 소스 영상 데이터로부터 청크 데이터(144)를 추출하고, 추출된 청크 데이터(144)를 제2 매칭 검사부(342)로 출력한다(제320 단계).

<132> 제320 단계후에, 제2 매칭 검사부(342)는 데이터 추출부(340)에서 추출된 청크 데

이타(144)가 소정 초기값을 포함하고 있는가 그렇지 않으면 매칭 체인의 i를 포함하고 있는가를 검사하고, 검사된 결과를 제2 매칭 신호로서 제1 및 제2 체인 복원부들(344 및 346)로 각각 출력한다(제322 단계).

<133> 만일, 청크 데이터가 소정 초기값을 포함하고 있으면, 예를 들면 청크 데이터의 제1 바이트에서 $i=0$ 이면, 청크 데이터에 포함된 제2 카운팅 값(count2)을 복호화하고, 복호화된 결과 및 압축된 소스 영상 데이터로부터 추출한 체인 데이터를 이용하여 매칭 체인을 갖지 않는 압축 체인을 복원한다(제324 단계). 이를 위해, 제1 체인 복원부(344)는 제2 매칭 신호를 통해 청크 데이터에 소정 초기값이 포함되어 있음을 인지하고, 데이터 추출부(340)로부터 출력되는 청크 데이터에 포함된 제2 카운팅 값(count2)을 복호화하고, 복호화된 결과 및 입력단자 IN5를 통해 입력한 압축된 소스 영상 데이터로부터 추출한 체인 데이터를 이용하여 매칭 체인을 갖지 않는 압축 체인을 복원하여 제2 데이터 출력부(348)로 출력한다. 제324 단계는 도 11에 도시된 제96 단계를 역으로 수행한다.

<134> 그러나, 청크 데이터가 매칭 체인의 i를 갖고 있으면, 예를 들면 청크 데이터의 제1 바이트에서 $i \neq 0$ 이면, 청크 데이터를 복호화하고, 복호화된 청크 데이터를 이용하여 매칭 체인을 갖는 압축 체인을 복원한다(제326 단계). 이를 위해, 제2 체인 복원부(346)는 제2 매칭 신호를 통해 청크 데이터에 매칭 체인의 i가 포함되어 있음을 인지하고, 데이터 추출부(340)로부터 입력한 청크 데이터를 복호화하고, 복호화된 청크 데이터를 이용하여 매칭 체인을 갖는 압축 체인을 복원하여 제2 데이터 출력부(348)로 출력한다. 제326 단계는 도 11에 도시된 제94 단계를 역으로 수행한다.

<135> 제324 또는 제326 단계후에, 제2 제어부(350)는 데이터 추출부(340)로부터 입력한 청크 데이터가 마지막 청크 데이터인가를 검사하고, 검사된 결과를 제2 제어 신호(C2)로

서 데이터 추출부(340)로 출력한다(제328 단계). 그러므로, 데이터 추출부(340)는 제2 제어 신호(C2)를 통해 체크 데이터가 마지막 체크 데이터가 아닌 것으로 인지되면, 입력 단자 IN5를 통해 입력한 압축된 소스 영상 데이터로부터 새로운 체크 데이터를 추출한다. 그러나, 데이터 추출부(340)는 제2 제어 신호(C2)를 통해 체크 데이터가 마지막 체크 데이터인 것으로 인지되면 입력단자 IN5를 통해 입력한 압축된 소스 영상 데이터로부터 체크 데이터를 더 이상 추출하지 않고, 제2 데이터 출력부(348)는 제2 제어 신호(C2)를 통해 체크 데이터가 마지막 체크 데이터인 것으로 인지되면 제1 및 제2 체인 복원부들(344 및 346)로부터 출력되는 복원된 결과들을 합성하여 엔트로피 디코딩된 결과로서 출력단자 OUT3을 통해 제2 데이터 치환부(314)로 출력한다(제330 단계).

<136> 한편, 제302 단계후에, 제2 데이터 치환부(314)는 엔트로피 디코더(312)로부터 출력되는 엔트로피 디코딩된 결과의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 복원된 소스 영상 데이터로서 출력단자 OUT2를 통해 출력한다(제304 단계). 제304 단계는 도 1에 도시된 제10 단계와 동일하므로 상세한 설명은 생략한다.

<137> 도 30은 전술한 본 발명에 의한 데이터 압축 및 복원 장치들을 포함하는 시스템을 나타내는 도면으로서, 개인용 컴퓨터(360), 입/출력(I/O: Input/Output) 포트(port)(364) 및 하드 카피 장치(366)로 구성된다.

<138> 도 30을 참조하면, 개인용 컴퓨터(360)는 하드 카피를 위한 데이터를 데이터 압축 장치(362)로 출력한다. 개인용 컴퓨터(360)에 내장되는(embedded) 본 발명에 의한 데이터 압축 장치(362)는 하드 카피를 위한 소스 영상 데이터를 입력하여 전술한 본 발명에 의한 데이터 압축 방법에 의해 압축하고, 압축된 결과를 I/O port(364)로 출력한다. I/O port(364)는 데이터 압축 장치(362)로부터 입력한 압축된 소스 영상 데이터를 하드 카피

장치(366)로 보내준다.

<139> 하드 카피 장치(366)에 내장되는(embedded) 본 발명에 의한 데이터 복원 장치(368)는 I/O port(364)를 통해 데이터 압축 장치(362)로부터 입력한 압축된 소스 영상 데이터를 원래의 소스 영상 데이터로 복원하고, 복원된 소스 영상 데이터를 하드 카피 장치(366)로 출력한다. 이 때, 하드 카피 장치(366)는 복원된 소스 영상 데이터를 하드 카피한다.

【발명의 효과】

<140> 이상에서 설명한 바와 같이, 본 발명에 의한 하드 카피 장치용 데이터 압축 및 복원 방법들 및 장치들은 소스 영상 데이터를 엔트로피 디코딩하기 전에 치환하기 때문에 압축율을 개선시킬 수 있고, 템플릿의 크기를 자유롭게 조절하여 압축 속도도 개선시킬 수 있는 효과를 갖는다.

【특허청구범위】

【청구항 1】

두 가지의 단계들로 촬영된 영상을 하드 카피하기 위해 사용되며, 메모리에 바이트 단위로 저장되는 소스 영상 데이터를 압축하는 하드 카피 장치용 데이터 압축 방법에 있어서,

(a) 상기 소스 영상 데이터의 열 바이트들을 행 바이트들로 치환하는 단계; 및

(b) 현재 압축 처리할 현재 체인의 값과 동일한 값을 갖는 체인이 이전에 압축 처리된 이전 체인들로 이루어진 사전에 존재하는가에 따라 상기 현재 체인과 상기 현재 체인에 후속하는 체인(들)로 이루어진 순차적 체인들 또는 상기 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 상기 소스 영상 데이터를 압축한 결과로서 결정하는 단계를 구비하고,

상기 행 바이트들중 이웃하는 행 바이트들은 이웃하는 메모리 어드레스를 갖고, 상기 열 바이트들중 이웃하는 열 바이트들의 오프셋은 상기 소스 영상 데이터의 행 폭에 해당하고, 상기 압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보 및 헤더에 대한 정보는 상기 (b) 단계가 수행되기 이전에 결정되고 상기 압축한 결과에 포함되며, 상기 체인은 적어도 둘 이상의 연속하는 상기 행 바이트들로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 2】

제1 항에 있어서, 상기 (b) 단계는

(b1) 상기 현재 체인의 값을 읽는 단계;

(b2) 상기 (b1) 단계에서 읽혀진 값과 동일한 값을 갖는 체인인 매칭 체인이 상기 사전에 존재하는가를 판단하는 단계;

(b3) 상기 매칭 체인이 상기 사전에 존재하면, 상기 매칭 체인의 i (여기서, i 는 상기 사전에 포함된 체인들 각각을 표시하는 첨자를 나타낸다.) 및 상기 매칭 체인을 갖는 체인들이 순차되는 개수를 카운팅하고, 카운팅된 결과인 제1 카운팅 값(count1)을 부호화하는 단계;

(b4) 상기 매칭 체인이 상기 사전에 존재하지 않으면, 상기 매칭 체인을 갖지 못하는 체인들이 순차되는 개수를 카운팅하여 제2 카운팅 값(count2)을 구하고, 상기 현재 체인의 값을 저장하는 단계;

(b5) 상기 (b3) 또는 상기 (b4) 단계후에, 상기 현재 체인이 마지막으로 압축 처리할 체인인가를 판단하는 단계;

(b6) 상기 현재 체인이 마지막으로 압축 처리할 체인이 아니면, 상기 현재 체인에 후속하는 체인을 새로운 상기 현재 체인으로 결정하고, 상기 (b1) 단계로 진행하는 단계;

(b7) 상기 현재 체인이 마지막으로 압축 처리할 체인이면, 상기 제2 카운팅 값에 따라 상기 소정 초기값 및 상기 제2 카운팅 값을 부호화하는 단계; 및

(b8) 상기 (b3) 및 상기 (b7) 단계들에서 부호화된 결과, 상기 시작 정보 및 상기 (b4) 단계에서 저장되는 상기 현재 체인의 값을 합성하여 상기 압축된 결과를 생성하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 3】

제1 항에 있어서, 상기 체인은 2개의 상기 행 바이트들로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 4】

제1 항에 있어서, 상기 체인은 4개의 상기 행 바이트들로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 5】

제2 항에 있어서, 상기 사전은 상기 이전 체인들 전부로 이루어진 서라운드에 해당하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 6】

제2 항에 있어서, 상기 하드 카피 장치용 데이터 압축 방법은

상기 (a) 단계후에 상기 이전 체인들중 상기 현재 체인과 유사하거나 동일한 값을 갖는 체인들로 이루어진 템플릿을 결정하고 상기 (b)단계로 진행하는 (c) 단계를 더 구비하고, 상기 사전은 상기 템플릿에 해당하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 7】

제6 항에 있어서, 상기 (c) 단계는

(c1) 상기 (a) 단계에서 치환된 상기 소스 영상 데이터에 적용된 의사 무작위 체인들 각각에 대한 서라운드를 구하는 단계;

(c2) 각각의 상기 서라운드에서 해당하는 상기 의사 무작위 체인의 값과 동일한 값을 갖는 체인(들)을 카운팅하는 단계;

(c3) 상기 서라운드에 포함된 체인들 각각을 표시하는 인덱스별로 상기 (c2) 단계에서 카운팅된 값을 분류하는 단계; 및

(c4) 분류된 카운팅된 값들중 가장 큰 상기 카운팅된 값부터 소정 개수(m)(여기서, m 은 상기 서라운드에 포함된 체인의 개수 이하이다.)의 카운팅된 값들을 선택하는 단계를 구비하고,

상기 서라운드는 상기 의사 무작위 체인이 압축 처리되기 이전에 압축 처리된 체인들로 이루어지고, 상기 템플릿은 상기 선택된 카운팅된 값들에 해당하는 인덱스들이 표시하는 체인들로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 8】

제7 항에 있어서, 상기 시작 정보는 상기 템플릿에 포함되는 체인들중 최초에 압축 처리되는 체인으로서 히스토리를 갖는 체인에 대한 정보를 가지고 있으며, 상기 히스토리는 상기 이전 체인(들)로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 9】

제7 항에 있어서, 상기 시작 정보는 상기 템플릿에 포함되는 체인들중 가장 큰 $T[i]$ (여기서, $1 \leq i \leq m$ 이고, $T[i]$ 는 상기 현재 체인과 i 가 표시하는 체인 사이의 거리를 나타낸다.)를 갖는 체인에 대한 정보를 갖는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 10】

제8 항에 있어서, 상기 템플릿에 포함되는 체인들중 값($c-T[i]$)(여기서, c 는 상기 현재 체인의 인덱스를 나타내고, $1 \leq i \leq m$ 을 나타내고, $T[i]$ 는 상기 현재 체인과 i 가 표시하는 체인 사이의 거리를 나타낸다.)이 음이 아닌 값을 갖는 체인들만이 상기 사전에 포함되는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 11】

제2 항에 있어서, 상기 (b3) 단계는

서로 동일한 값을 갖는 체인들이 순차되는 개수를 카운팅하여 상기 제1 카운팅 값을 구하고, 상기 제1 카운팅 값을 런 길이 부호화 방식에 의해 부호화하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 12】

제7 항에 있어서, 상기 (c8) 단계에서 생성한 상기 압축된 결과는

상기 (b3) 단계에서 부호화된 상기 i 및 상기 제1 카운팅 값, 또는 상기 (b7) 단계에서 부호화된 상기 소정 초기값 및 상기 제2 카운팅 값을 갖는 체크 데이터;

부호화되지 않은 상기 시작 정보 및 상기 (b4) 단계에서 저장된 부호화되지 않은 현재 체인의 값을 갖는 체인 데이터; 및

상기 체인 데이터의 크기 및 상기 사전의 길이를 갖는 상기 헤더를 구비하는 것을 특징으로 하고,

상기 체크 데이터는 단위 바이트로 정렬된(one-byte aligned) 데이터에 해당하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 13】

제12 항에 있어서, 상기 체크 데이터는

상기 매칭 체인의 i 또는 상기 소정 초기값을 표시하는 M (여기서, $m=2^M-1$) 비트들 및 상기 제1 또는 상기 제2 카운팅 값을 표시하는 $8-M$ 비트들을 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 14】

제13 항에 있어서, 상기 체크 데이터는 상기 제1 또는 상기 제2 카운팅 값만을 표시하는 비트들을 더 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 15】

제14 항에 있어서, 상기 M 은 3, 4, 5 또는 6으로 될 수 있는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 16】

제12 항에 있어서, 상기 (b3) 단계는

상기 매칭 체인이 상기 사전에 존재하면, 상기 제2 카운팅 값을 플러쉬하는 단계; 및

상기 매칭 체인의 상기 i 및 상기 제1 카운팅 값을 부호화하고, 상기 (b5) 단계로 진행하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 17】

제16 항에 있어서, 상기 (b4) 단계는

상기 매칭 체인이 상기 사전에 존재하지 않으면, 상기 현재 체인의 값을 상기 체인 데이터로 결정하여 저장하는 단계;

상기 제2 카운팅 값을 1만큼 증가시키는 단계;

상기 제2 카운팅 값이 소정 최대값인가를 판단하고, 상기 제2 카운팅 값이 상기 소정 최대값이 아니면, 상기 (b5) 단계로 진행하는 단계; 및

상기 제2 카운팅 값이 상기 소정 최대값이면, 상기 제2 카운팅 값을 플러쉬하고 상기 (b5) 단계로 진행하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 18】

제17 항에 있어서, 상기 (b7) 단계는

상기 현재 체인이 마지막으로 압축 처리할 체인이면, 상기 제2 카운팅 값이 소정 임계값보다 큰가를 판단하고, 상기 제2 카운팅 값이 상기 소정 임계값보다 크지 않으면 상기 데이터 압축 방법을 종료하는 단계; 및

상기 제2 카운팅 값이 상기 소정 임계값보다 크면, 상기 소정 초기값 및 상기 제2 카운팅 값을 부호화하고, 상기 데이터 압축 방법을 종료하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 방법.

【청구항 19】

두 가지의 단계들로 촬영된 영상을 하드 카피하기 위해 사용되며, 메모리에 바이트 단위로 저장되는 소스 영상 데이터를 압축하는 하드 카피 장치용 데이터 압축 장치에 있어서,

입력한 상기 소스 영상 데이터의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 출력하는 제1 데이터 치환부;

상기 제1 데이터 치환부로부터 출력되는 치환된 결과를 입력하여, 이전에 압축 처리된 이전 체인들중 현재 압축 처리할 현재 체인과 유사하거나 동일한 값을 갖는 체인들로 이루어진 템플릿을 결정하여 출력하는 템플릿 결정부; 및

압축 처리를 시작할 체인에 대한 정보를 갖는 시작 정보에 응답하여 상기 제1 데이터 치환부로부터 입력한 상기 현재 체인의 값과 동일한 값을 갖는 체인이 상기 템플릿 결정부로부터 입력한 템플릿에 존재하는가를 검사하고, 검사된 결과에 응답하여 상기 현재 체인과 상기 현재 체인에 후속하는 체인(들)로 이루어진 상기 순차적 체인들 또는 상기 현재 체인을 엔트로피 엔코딩하고, 엔트로피 엔코딩된 결과를 상기 소스 영상 데이터를 압축한 결과로서 출력하는 엔트로피 엔코더를 구비하고,

상기 행 바이트들중 이웃하는 행 바이트들은 이웃하는 메모리 어드레스를 갖고, 상기 열 바이트들중 이웃하는 열 바이트들의 오프셋은 상기 소스 영상 데이터의 행 폭에 해당하고, 상기 시작 정보에 대한 정보 및 헤더에 대한 정보는 미리 결정되어 상기 엔트로피 엔코더로 제공되고 상기 압축한 결과에 포함되고, 상기 체인은 적어도 둘 이상의 연속하는 상기 행 바이트들로 이루어지는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 장치.

【청구항 20】

제19 항에 있어서, 상기 엔트로피 엔코더는

상기 시작 정보와 상기 헤더에 대한 정보들 및 제1 제어 신호에 응답하여, 상기 제1 데이터 치환부로부터 입력한 상기 현재 체인의 값을 읽는 데이터 입력부;

상기 데이터 입력부에서 읽혀진 값과 동일한 값을 갖는 체인인 매칭 체인이 상기 템플릿 결정부로부터 입력한 상기 템플릿에 존재하는가를 검사하고, 검사된 결과로부터 매칭 체인의 유무를 나타내는 제1 매칭 신호 및 상기 매칭 체인의 i (여기서, i 는 상기 템플릿에 포함된 체인들 각각을 표시하는 첨자를 나타낸다.)를 검출하는 제1 매칭 검사부;

상기 제1 매칭 신호에 응답하여, 상기 매칭 체인을 갖는 체인들이 순차되는 개수를 카운팅하고, 카운팅된 결과인 제1 카운팅 값(count1)을 출력하는 제1 카운터;

상기 제1 매칭 신호에 응답하여, 상기 매칭 체인을 갖지 못하는 체인들이 순차되는 개수를 카운팅하고, 카운팅된 결과인 제2 카운팅 값(count2)을 출력하는 제2 카운터;

상기 제1 카운팅 값 및 상기 제1 매칭 검사부로부터 입력한 상기 매칭 체인의 i 를 부호화하고, 부호화된 결과를 출력하는 제1 부호화부;

상기 제1 제어 신호에 응답하여, 상기 제2 카운팅 값을 소정 임계값과 비교하고, 비교된 결과를 출력하는 비교부;

상기 비교부에서 비교된 결과에 응답하여, 외부로부터 입력한 소정 초기값 및 상기 제2 카운팅 값을 부호화하는 제2 부호화부;

상기 제1 매칭 신호에 응답하여, 상기 제1 및 상기 제2 부호화부에서 부호화된 결과들, 상기 제1 데이터 치환부로부터 입력한 상기 현재 체인의 값 및 외부로부터 입력한 상기 시작 정보를 저장하는 저장부;

상기 제1 데이터 치환부로부터 입력한 상기 현재 체인이 상기 소스 영상 데이터에서 마지막으로 압축 처리할 체인인가를 검사하고, 검사된 결과를 상기 제1 제어 신호로서 출력하는 제1 제어부; 및

상기 저장부로부터 입력한 상기 시작 정보, 상기 현재 체인의 값, 상기 제1 및 상기 제2 부호화부들에서 부호화된 결과들을 합성하고, 합성된 결과를 상기 압축된 결과로서 출력하는 제1 데이터 출력부를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 장치.

【청구항 21】

제20 항에 있어서, 상기 저장부는

상기 제1 매칭 신호에 응답하여, 상기 제1 부호화부에서 부호화된 결과 또는 상기 제2 부호화부에서 부호화된 결과를 선택적으로 체크 데이터로서 저장하는 제1 저장부; 및

상기 제1 매칭 신호에 응답하여, 상기 현재 체인의 값 및 상기 시작 정보를 체인 데이터로서 저장하는 제2 저장부를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 압축 장치.

【청구항 22】

제12 항에 있어서, 상기 압축된 소스 영상 데이터로부터 상기 소스 영상 데이터를 복원해내는 하드 카피 장치용 데이터 복원 방법에 있어서,

(d) 상기 압축된 소스 영상 데이터한 결과로부터 상기 헤더와 상기 시작 정보를 추출하는 단계;

(e) 상기 압축된 소스 영상 데이터에서 추출한 각 상기 청크 데이터 및 각 상기 체인 데이터에 포함된 압축 체인들을 상기 (d) 단계에서 추출된 상기 헤더 및 상기 시작 정보를 이용하여 엔트로피 디코딩하는 단계; 및

(f) 상기 엔트로피 디코딩된 결과의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 복원된 소스 영상 데이터로서 결정하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 복원 방법.

【청구항 23】

제22 항에 있어서, 상기 (e) 단계는

(e1) 상기 헤더 및 상기 시작 정보를 이용하여, 상기 압축된 소스 영상 데이터로부터 상기 청크 데이터를 추출하는 단계;

(e2) 상기 (e1) 단계에서 추출된 상기 청크 데이터가 상기 소정 초기값을 갖고 있는가 그렇지 않으면 상기 매칭 체인의 i 를 갖고 있는가를 판단하는 단계;

(e3) 상기 청크 데이터가 상기 소정 초기값을 갖고 있으면, 상기 청크 데이터에 포함된 상기 제2 카운팅 값을 복호화하고, 복호화된 결과 및 상기 압축된 소스 영상 데이터한 결과로부터 추출한 상기 체인 데이터를 이용하여 상기 매칭 체인을 갖지 않는 상기 압축 체인을 복원하는 단계;

(e4) 상기 청크 데이터가 상기 매칭 체인의 i 를 갖고 있으면, 상기 청크 데이터를 복호화하고, 복호화된 청크 데이터를 이용하여 상기 매칭 체인을 갖는 상기 압축 체인을 복원하는 단계;

(e5) 상기 청크 데이터가 상기 압축된 소스 영상 데이터에서 마지막 청크 데이터

인가를 판단하고, 상기 청크 데이터가 마지막 청크 데이터가 아니면, 상기 (e1) 단계로 진행하는 단계; 및

(e6) 상기 청크 데이터가 마지막 청크 데이터이면, 상기 (e3) 또는 상기 (e4) 단계에서 복원된 체인들을 합성하여, 상기 엔트로피 디코딩된 결과를 생성하는 단계를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 복원 방법.

【청구항 24】

제12 항에 있어서, 상기 압축된 소스 영상 데이터한 결과로부터 상기 소스 영상 데이터를 복원해내는 하드 카피 장치용 데이터 복원 장치에 있어서,

입력한 상기 압축된 소스 영상 데이터로부터 상기 헤더와 상기 시작 정보를 추출하는 정보 추출부;

상기 정보 추출부로부터 출력되는 상기 헤더 및 상기 시작 정보에 응답하여, 상기 압축된 소스 영상 데이터에서 추출한 각 상기 청크 데이터 및 각 상기 체인 데이터에 포함된 압축 체인들을 엔트로피 디코딩하고, 엔트로피 디코딩된 결과를 출력하는 엔트로피 디코더; 및

상기 엔트로피 디코더로부터 출력되는 상기 엔트로피 디코딩된 결과의 열 바이트들을 행 바이트들로 치환하고, 치환된 결과를 복원된 소스 영상 데이터로서 출력하는 제2 데이터 치환부를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 복원 장치.

【청구항 25】

제24 항에 있어서, 상기 엔트로피 디코더는

상기 정보 추출부로부터 출력되는 상기 헤더와 상기 시작 정보 및 제2 제어 신호

에 응답하여, 상기 압축된 소스 영상 데이터로부터 상기 청크 데이터를 추출하는 데이터 추출부;

상기 데이터 추출부에서 추출된 상기 청크 데이터가 상기 소정 초기값을 포함하고 있는가 그렇지 않으면 상기 매칭 체인의 i 를 포함하고 있는가를 검사하고, 검사된 결과를 제2 매칭 신호로서 출력하는 제2 매칭 검사부;

상기 제2 매칭 신호에 응답하여, 상기 청크 데이터에 포함된 상기 제2 카운팅 값을 복호화하고, 복호화된 결과 및 상기 압축된 소스 영상 데이터로부터 추출한 상기 체인 데이터를 이용하여 상기 매칭 체인을 갖지 않는 상기 압축 체인을 복원하여 출력하는 제1 체인 복원부;

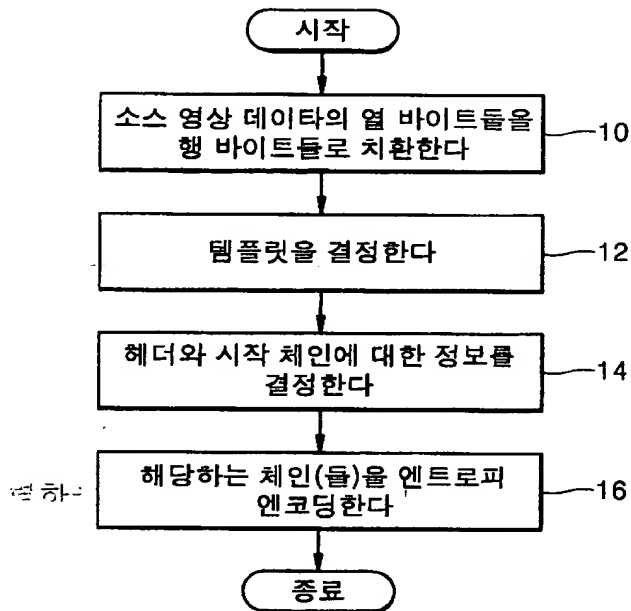
상기 제2 매칭 신호에 응답하여, 상기 청크 데이터를 복호화하고, 복호화된 청크 데이터를 이용하여 상기 매칭 체인을 갖는 상기 압축 체인을 복원하여 출력하는 제2 체인 복원부;

상기 데이터 추출부로부터 입력한 상기 청크 데이터가 상기 압축된 소스 영상 데이터에서 마지막 청크 데이터인가를 검사하고, 검사된 결과를 상기 제2 제어 신호로서 출력하는 제2 제어부; 및

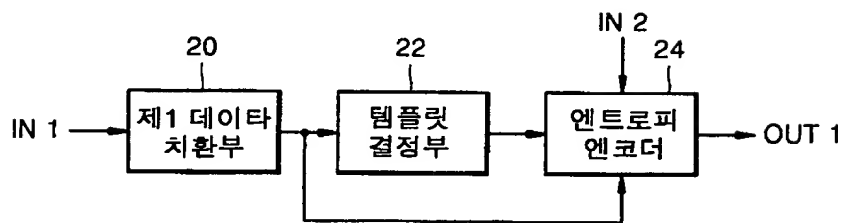
상기 제2 제어 신호에 응답하여, 상기 제1 및 상기 제2 체인 복원부들로부터 출력되는 복원된 결과들을 합성하여 상기 엔트로피 디코딩된 결과로서 출력하는 제2 데이터 출력부를 구비하는 것을 특징으로 하는 하드 카피 장치용 데이터 복원 장치.

【도면】

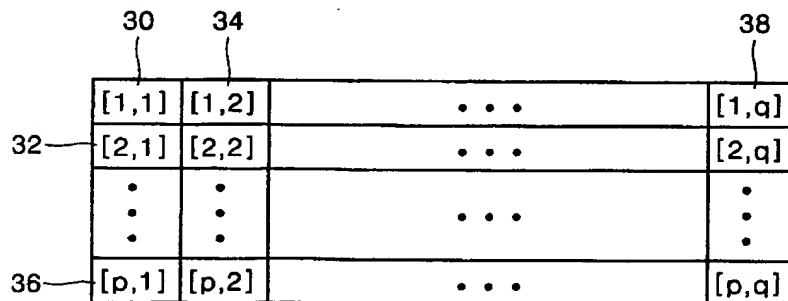
【도 1】



【도 2】



【도 3】



【도 4】

[1,1]	[2,1]	...	[p,1]
[1,2]	[2,2]	...	[p,2]
⋮	⋮	...	⋮
[1,q]	[2,q]	...	[p,q]

【도 5】

[illegible]

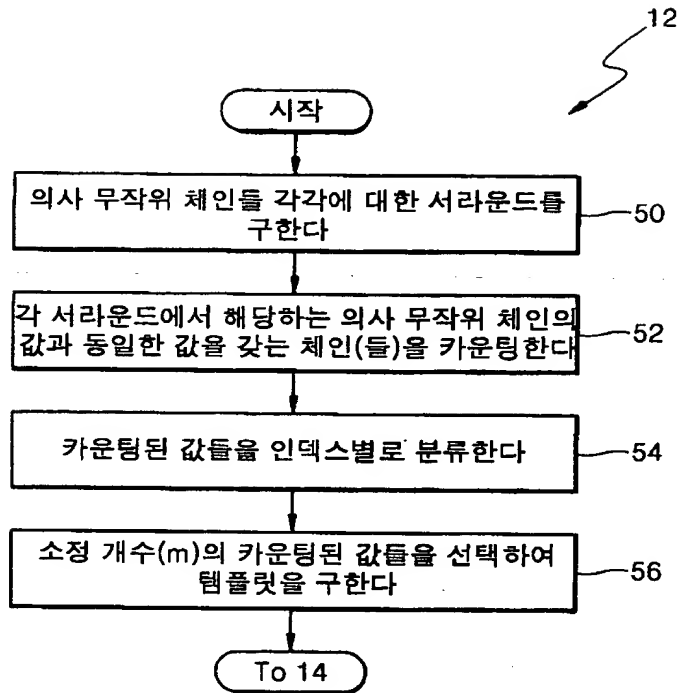
【도 6】

[illegible]

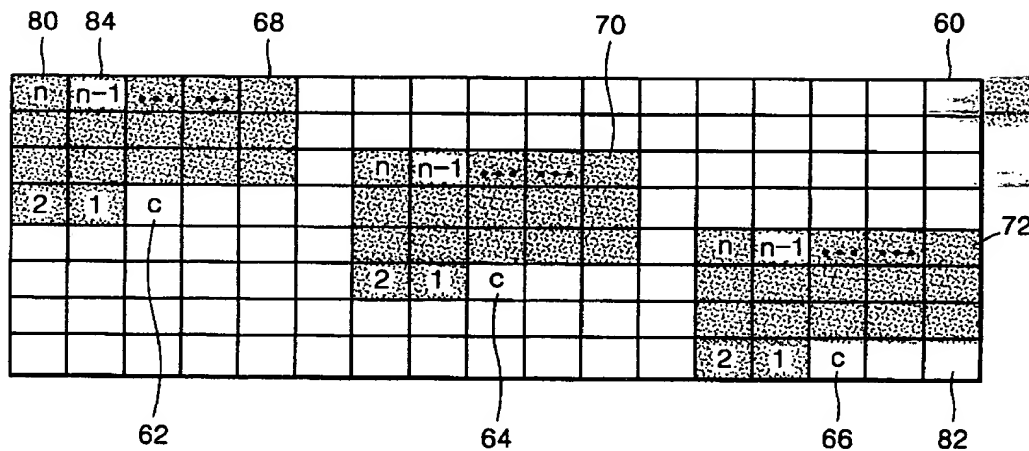
【도 7】

[illegible]

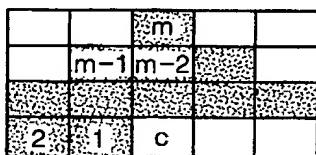
【도 8】



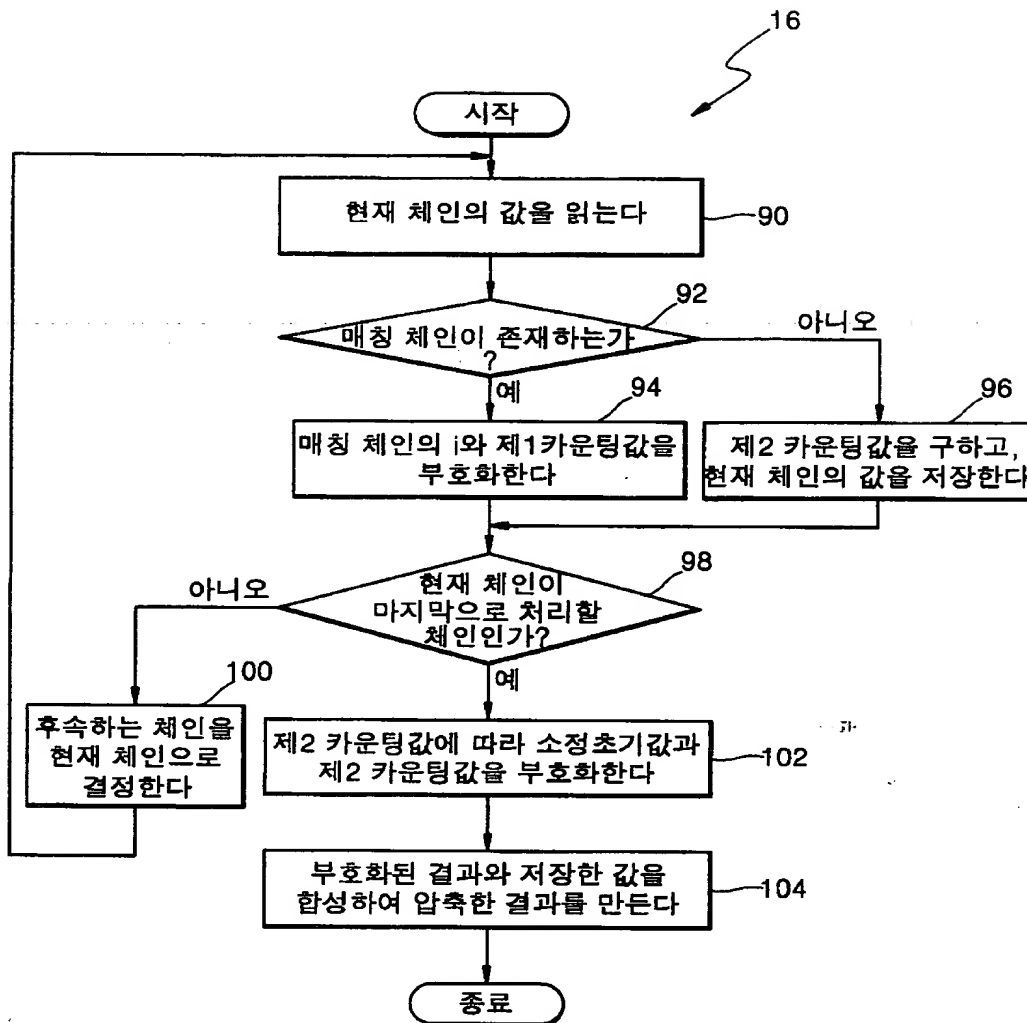
【도 9】



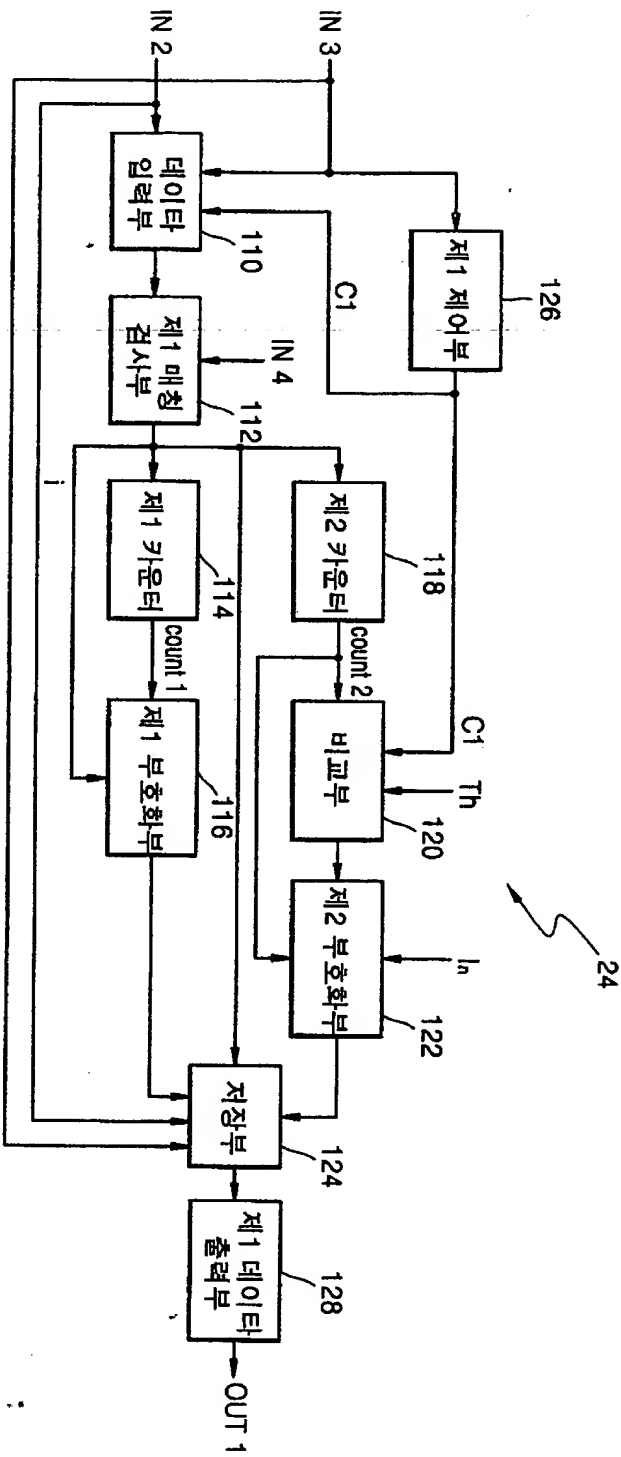
【도 10】



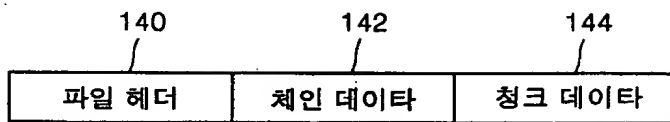
【도 11】



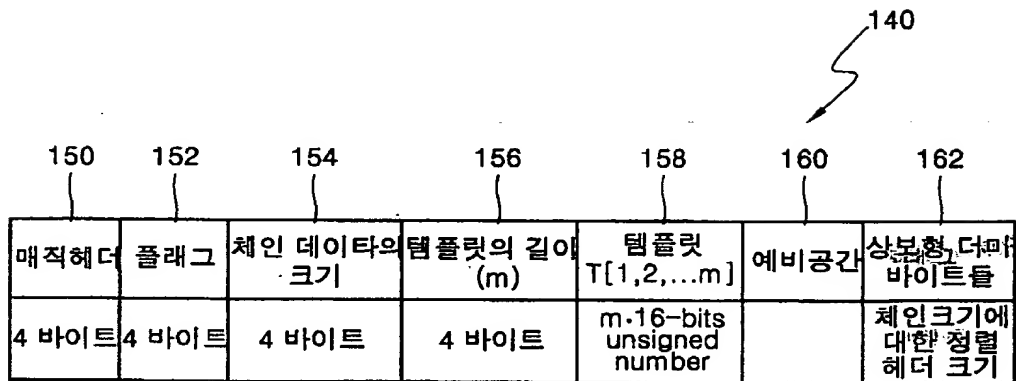
【도 12】



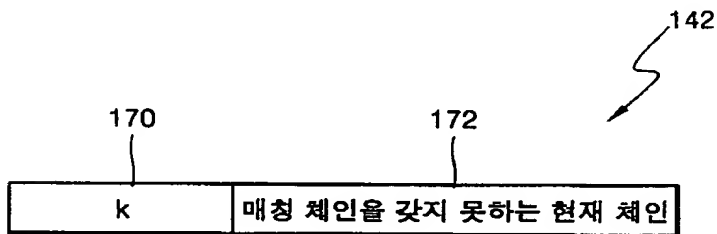
【도 13】



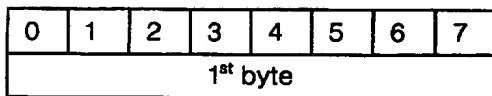
【도 14】



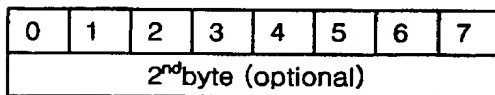
【도 15】



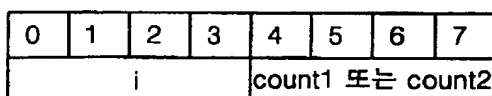
【도 16】



【도 17】



【도 18a】



【도 18b】

0	1	2	3	4	5	6	7
count A (optional)							

【도 19a】

0	1	2	3	4	5	6	7
i					count1 또는 count2		

【도 19b】

0	1	2	3	4	5	6	7
count A (optional)							

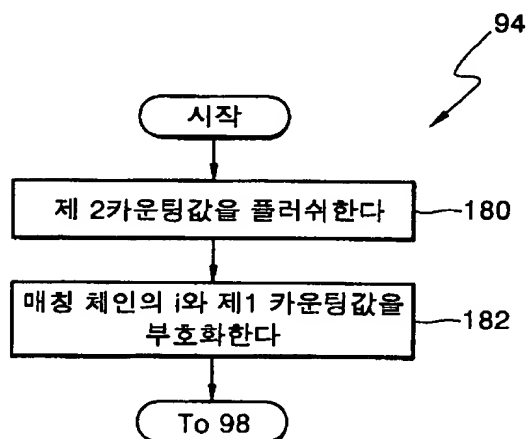
【도 20a】

0	1	2	3	4	5	6	7
i					count1 또는 count2		

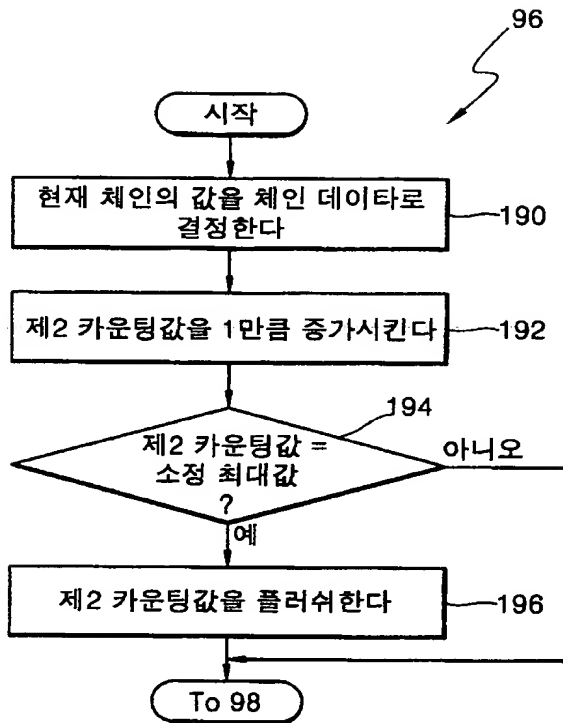
【도 20b】

0	1	2	3	4	5	6	7
count A (optional)							

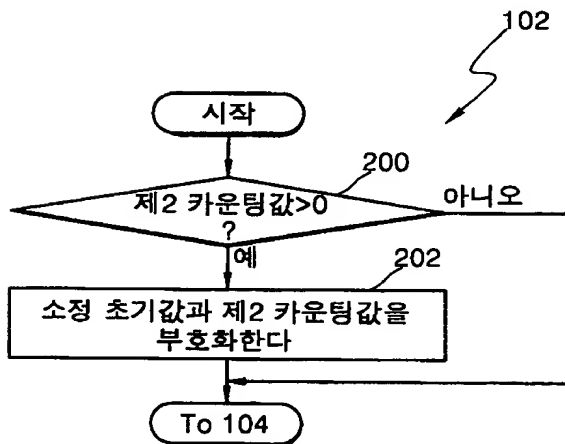
【도 21】



【도 22】



【도 23】



【도 24】

				17			
		16	15	14	13	12	
	11	10	9	8	7	6	5
4	3	2	1	c			

【도 25a】

X	X	X	X	X	X	X	X
X	X	X	X	X	A	B	C
X	X	A	X	A	B	X	X
X	X	X	A	A	B	C	Y

220 222 224
210 212 214

【도 25b】

X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X
X	X	X	A	A	A	A	X

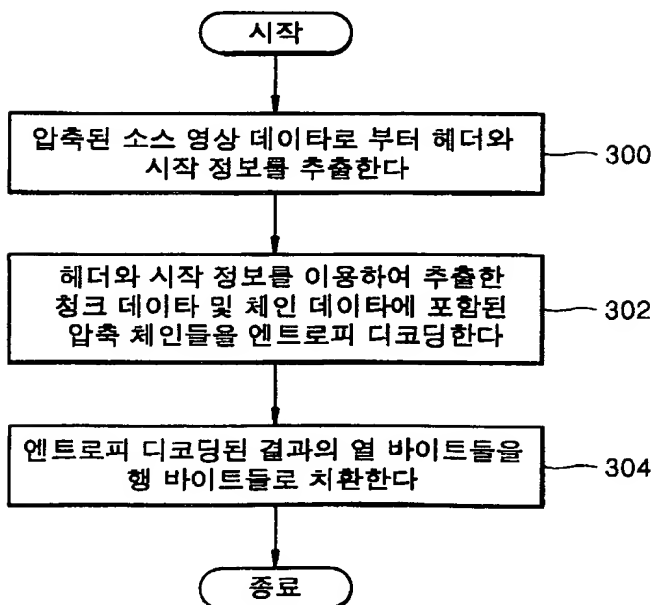
236 230 232 234

【도 25c】

X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X
X	X	X	X	A	B	C	X

240 242 244

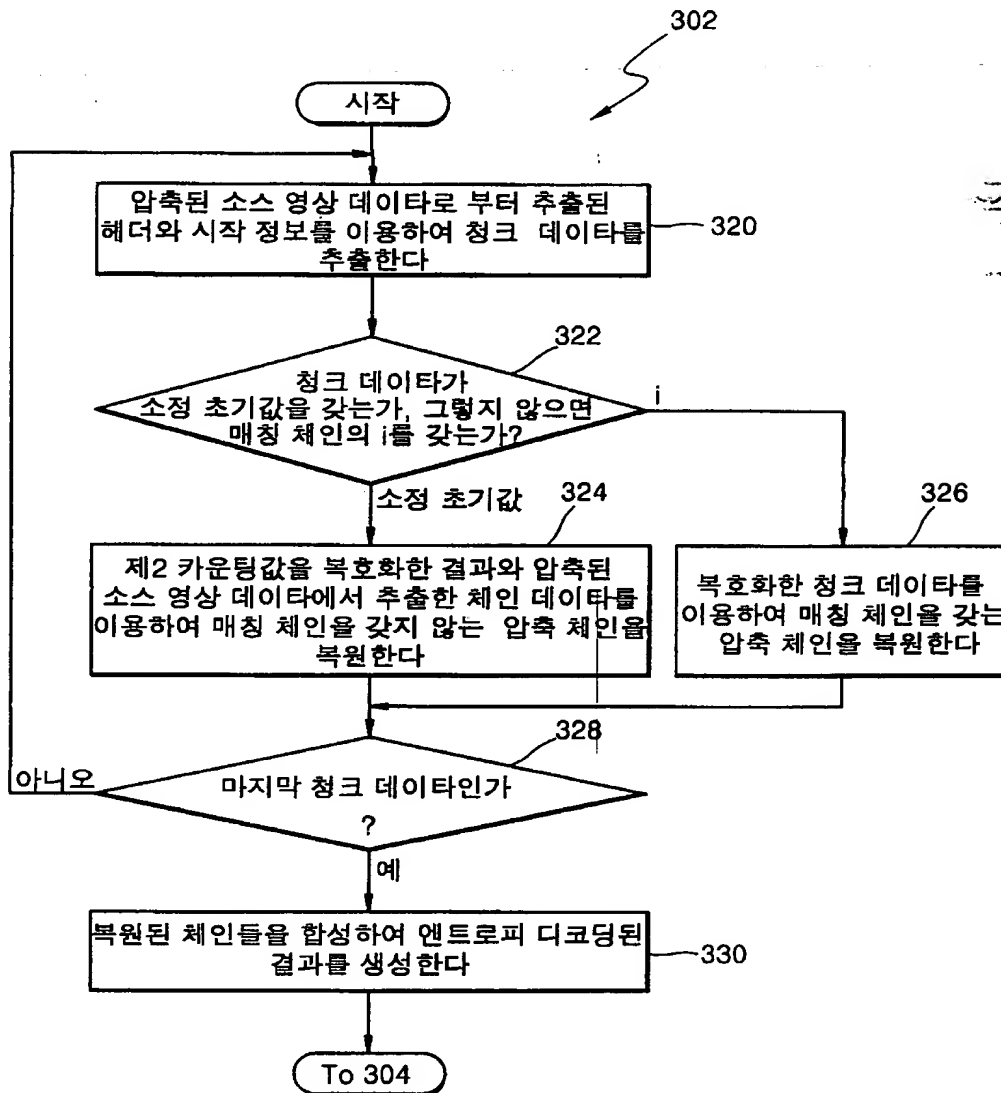
【도 26】



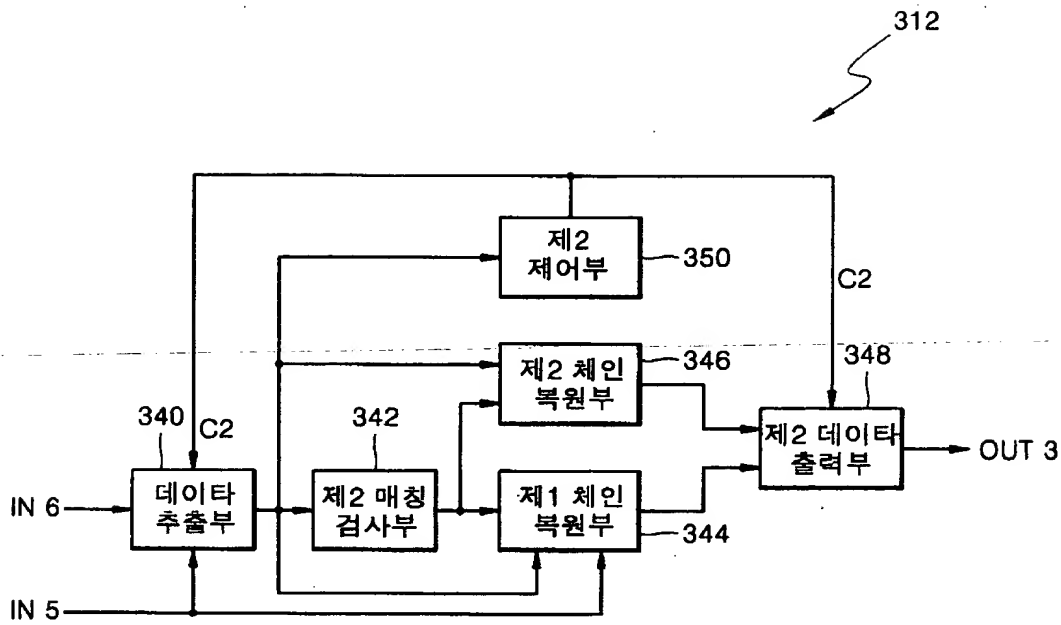
【도 27】



【도 28】



【도 29】



【도 30】

